

# OpenStack

## Networking API v2.0

(February 6, 2017)



[docs.openstack.org](https://docs.openstack.org)

# Networking API v2.0

- General API Overview
  - API guide
  - API versions
  - Extensions
- Layer 2 Networking
  - Networks
  - Ports
  - Segments
  - Trunk networking
  - Trunk details extended attributes (ports)
- Layer 3 Networking
  - Floating IPs (floatingips)
  - Routers (routers)
  - Subnet pools extension (subnetpools)
  - Subnets
- Security
  - FWaaS v1.0 (DEPRECATED) (fw, firewalls, firewall\_policies, firewall\_rules)
  - FWaaS v2.0 (CURRENT) (fwaas, firewall\_groups, firewall\_policies, firewall\_rules)
  - Security group rules (security-group-rules)
  - Security groups (security-groups)
  - VPNaaS 2.0 (UNMAINTAINED) (vpn, vpnservices, ikepolicies, ipsecpolicies, endpoint-groups, ipsec-site-connections)
- Resource Management
  - Networking Flavors Framework v2.0 (CURRENT) (flavor, service\_profile)
  - Metering labels and rules (metering-labels, metering-label-rules)
  - Network IP availability and usage stats
  - Quotas extension (quotas)
  - Service providers
  - Tag extension (tags)
- Quality of Service
  - QoS policies (qos)
- Load Balancer as a Service
  - LBaaS 2.0 (STABLE)
  - LBaaS 1.0 (DEPRECATED)
- Logging Resource (networking-midonet)
  - Logging Resources
  - Firewall Logs
- Router interface floating IP (networking-midonet)
  - Router Interface floating IP
- FIP64 (networking-midonet)
  - FIP64

# General API Overview

## API guide

This section introduces readers to OpenStack Networking (v2) API, providers guidelines on how to use it, and describes common features available to users throughout all Networking APIs.

### General information

The Networking API v2.0 is a ReSTful HTTP service that uses all aspects of the HTTP protocol including methods, URIs, media types, response codes, and so on. Providers can use existing features of the protocol including caching, persistent connections, and content compression. For example, providers who employ a caching layer can respond with a 203 code instead of a 200 code when a request is served from the cache. Additionally, providers can offer support for conditional **GET** requests by using ETags, or they may send a redirect in response to a **GET** request. Create clients so that these differences are accounted for.

### Authentication and authorization

The Networking API v2.0 uses the [OpenStack Identity service](#) as the default authentication service. When Keystone is enabled, users that submit requests to the OpenStack Networking service must provide an authentication token in **X-Auth-Token** request header. You obtain the token by authenticating to the Keystone endpoint.

When Keystone is enabled, the `project_id` attribute is not required in create requests because the project ID is derived from the authentication token.

NOTE: Currently the Networking API accepts the deprecated `tenant_id` attribute for the project ID for backward compatibility.

The default authorization settings allow only administrative users to create resources on behalf of a different project.

OpenStack Networking uses information received from Keystone to authorize user requests. OpenStack Networking handles the following types of authorization policies:

- **Operation-based policies** specify access criteria for specific operations, possibly with fine-grained control over specific attributes.
- **Resource-based policies** access a specific resource. Permissions might or might not be granted depending on the permissions configured for the resource. Currently available for only the network resource.

The actual authorization policies enforced in OpenStack Networking might vary from deployment to deployment.

## Request and response formats

The Networking API v2.0 supports JSON data serialization request and response formats only.

### Request format

The Networking API v2.0 only accepts requests with the JSON data serialization format. The Content-Type header is ignored.

### Tenant and project attributes in requests

Starting with the Newton release of the Networking service, the Networking API accepts the `project_id` attribute in addition to the `tenant_id` attribute in requests. The `tenant_id` attribute is accepted for backward compatibility. If both the `project_id` and the `tenant_id` attribute are provided in the same request, their values must be identical.

To determine whether a Networking API v2.0 endpoint supports the `project_id` attribute in requests, check that the `project-id` API extension is enabled (see [Extensions](#)).

### Response format

The Networking API v2.0 always responds with the JSON data serialization format. The Accept header is ignored.

### Query extension

A `.json` extension can be added to the request URI. For example, the `.json` extension in the following requests are equivalent:

- **GET** *publicURL/networks*
- **GET** *publicURL/networks.json*

### Tenant and project attributes in responses

Starting with the Newton release of the Networking service, the Networking API returns a `project_id` attribute in responses, while still returning a `tenant_id` attribute for backward compatibility. The values will always be identical.

To determine whether a Networking API v2.0 endpoint returns the `project_id` attribute in responses, check that the `project-id` API extension is enabled (see [Extensions](#)).

## Filtering and column selection

The Networking API v2.0 supports filtering based on all top level attributes of a resource. Filters are applicable to all list requests.

For example, the following request returns all networks named foobar:

```
GET /v2.0/networks?name=foobar
```

When you specify multiple filters, the Networking API v2.0 returns only objects that meet all filtering criteria. The operation applies an AND condition among the filters.



## Note

OpenStack Networking does not offer an OR mechanism for filters.

Alternatively, you can issue a distinct request for each filter and build a response set from the received responses on the client-side.

By default, OpenStack Networking returns all attributes for any show or list call. The Networking API v2.0 has a mechanism to limit the set of attributes returned. For example, return `id`.

You can use the `fields` query parameter to control the attributes returned from the Networking API v2.0.

For example, the following request returns only `id` and `name` for each network:

```
GET /v2.0/networks.json?fields=id&fields=name
```

## Synchronous versus asynchronous plug-in behavior

The Networking API v2.0 presents a logical model of network connectivity consisting of networks, ports, and subnets. It is up to the OpenStack Networking plug-in to communicate with the underlying infrastructure to ensure packet forwarding is consistent with the logical model. A plug-in might perform these operations asynchronously.

When an API client modifies the logical model by issuing an HTTP **POST**, **PUT**, or **DELETE** request, the API call might return before the plug-in modifies underlying virtual and physical switching devices. However, an API client is guaranteed that all subsequent API calls properly reflect the changed logical model.

For example, if a client issues an HTTP **PUT** request to set the attachment for a port, there is no guarantee that packets sent by the interface named in the attachment are forwarded immediately when the HTTP call returns. However, it is guaranteed that a subsequent HTTP **GET** request to view the attachment on that port returns the new attachment value.

You can use the `status` attribute with the network and port resources to determine whether the OpenStack Networking plug-in has successfully completed the configuration of the resource.

## Bulk-create

The Networking API v2.0 enables you to create several objects of the same type in the same API request. Bulk create operations use exactly the same API syntax as single create operations except that you specify a list of objects rather than a single object in the request body.

Bulk operations are always performed atomically, meaning that either all or none of the objects in the request body are created. If a particular plug-in does not support atomic operations, the Networking API v2.0 emulates the atomic behavior so that users can expect the same behavior regardless of the particular plug-in running in the background.

OpenStack Networking might be deployed without support for bulk operations and when the client attempts a bulk create operation, a 400 Bad request error is

returned.

## Pagination

To reduce load on the service, list operations will return a maximum number of items at a time. To navigate the collection, the parameters `limit`, `marker` and `page_reverse` can be set in the URI. For example:

```
?limit=100&marker=1234&page_reverse=False
```

The `marker` parameter is the ID of the last item in the previous list. The `limit` parameter sets the page size. The `page_reverse` parameter sets the page direction. These parameters are optional. If the client requests a limit beyond the maximum limit configured by the deployment, the server returns the maximum limit number of items.

For convenience, list responses contain atom “next” links and “previous” links. The last page in the list requested with ‘`page_reverse=False`’ will not contain “next” link, and the last page in the list requested with ‘`page_reverse=True`’ will not contain “previous” link. The following examples illustrate two pages with three items. The first page was retrieved through:

```
GET http://127.0.0.1:9696/v2.0/networks.json?limit=2
```

Pagination is an optional feature of OpenStack Networking API, and it might be disabled. If pagination is disabled, the pagination parameters will be ignored and return all the items.

If a particular plug-in does not support pagination operations, and pagination is enabled, the Networking API v2.0 will emulate the pagination behavior so that users can expect the same behavior regardless of the particular plug-in running in the background.

To determine if pagination is supported, a user can check whether the ‘pagination’ extension API is available.

### Example Network collection, first page: JSON request

```
GET /v2.0/networks.json?limit=2 HTTP/1.1
Host: 127.0.0.1:9696
Content-Type: application/json
Accept: application/json
```

### Example Network collection, first page: JSON response

```
{
  "networks": [
    {
      "admin_state_up": true,
      "id": "396f12f8-521e-4b91-8e21-2e003500433a",
      "name": "net3",
      "provider:network_type": "vlan",
      "provider:physical_network": "physnet1",
      "provider:segmentation_id": 1002,
      "router:external": false,
      "shared": false,
      "status": "ACTIVE",
      "subnets": [],
      "tenant_id": "20bd52ff3e1b40039c312395b04683cf",
      "project_id": "20bd52ff3e1b40039c312395b04683cf"
    },
    {
      "admin_state_up": true,
      "id": "71c1e68c-171a-4aa2-aca5-50ea153a3718",
      "name": "net2",
      "provider:network_type": "vlan",
```

```

        "provider:physical_network": "physnet1",
        "provider:segmentation_id": 1001,
        "router:external": false,
        "shared": false,
        "status": "ACTIVE",
        "subnets": [],
        "tenant_id": "20bd52ff3e1b40039c312395b04683cf",
        "project_id": "20bd52ff3e1b40039c312395b04683cf"
    }
],
"networks_links": [
    {
        "href": "http://127.0.0.1:9696/v2.0/networks.json?limit=2&marker=71c1e68c-171a-4aa2-aca5-50ea153a3718",
        "rel": "next"
    },
    {
        "href": "http://127.0.0.1:9696/v2.0/networks.json?limit=2&marker=396f12f8-521e-4b91-8e21-2e003500433a&page_reverse=True",
        "rel": "previous"
    }
]
}

```

The last page won't show the "next" links

## Example Network collection, last page: JSON request

```

GET /v2.0/networks.json?limit=2&marker=71c1e68c-171a-4aa2-aca5-50ea153a3718 HTTP/1.1
Host: 127.0.0.1:9696
Content-Type: application/json
Accept: application/json

```

## Example Network collection, last page: JSON response

```

{
  "networks": [
    {
      "admin_state_up": true,
      "id": "b3680498-03da-4691-896f-ef9ee1d856a7",
      "name": "net1",
      "provider:network_type": "vlan",
      "provider:physical_network": "physnet1",
      "provider:segmentation_id": 1000,
      "router:external": false,
      "shared": false,
      "status": "ACTIVE",
      "subnets": [],
      "tenant_id": "c05140b3dc7c4555afff9fab6b58edc2",
      "project_id": "c05140b3dc7c4555afff9fab6b58edc2"
    }
  ],
  "networks_links": [
    {
      "href": "http://127.0.0.1:9696/v2.0/networks.json?limit=2&marker=b3680498-03da-4691-896f-ef9ee1d856a7&page_reverse=True",
      "rel": "previous"
    }
  ]
}

```

## Sorting

You can use the `sort_key` and `sort_dir` parameters to sort the results of list operations. Currently sorting does not work with extended attributes of resource. The `sort_key` and `sort_dir` can be repeated, and the number of `sort_key` and `sort_dir` provided must be same. The `sort_dir` parameter indicates in which direction to sort. Acceptable values are `asc` (ascending) and `desc` (descending).

Sorting is optional feature of OpenStack Networking API, and it might be disabled. If sorting is disabled, the sorting parameters are ignored.

If a particular plug-in does not support sorting operations and sorting is enabled, the Networking API v2.0 emulates the sorting behavior so that users can expect the same behavior regardless of the particular plug-in that runs in the background.

To determine if sorting is supported, a user can check whether the 'sorting' extension API is available.

## Extensions

The Networking API v2.0 is extensible.

The purpose of Networking API v2.0 extensions is to:

- Introduce new features in the API without requiring a version change.
- Introduce vendor-specific niche functionality.
- Act as a proving ground for experimental functionalities that might be included in a future version of the API.

To programmatically determine which extensions are available, issue a **GET** request on the **v2.0/extensions** URI.

To query extensions individually by unique alias, issue a **GET** request on the `/v2.0/extensions/*alias_name*` URI. Use this method to easily determine if an extension is available. If the extension is not available, a 404 Not Found response is returned.

You can extend existing core API resources with new actions or extra attributes. Also, you can add new resources as extensions. Extensions usually have tags that prevent conflicts with other extensions that define attributes or resources with the same names, and with core resources and attributes. Because an extension might not be supported by all plug-ins, the availability of an extension varies with deployments and the specific plug-in in use.

## Faults

The Networking API v2.0 returns an error response if a failure occurs while processing a request. OpenStack Networking uses only standard HTTP error codes. *4nn* errors indicate problems in the particular request being sent from the client.

Error	Description
400	Bad request Malformed request URI or body requested admin state invalid Invalid values entered Bulk operations disallowed Validation failed Method not allowed for request body (such as trying to update attributes that can be specified at create-time only)
404	Not Found Non existent URI Resource not found
409	Conflict Port configured on network IP allocated on subnet Conflicting IP allocation pools for subnet
500	Internal server error Internal OpenStack Networking error
503	Service unavailable Failure in Mac address generation

Users submitting requests to the Networking API v2.0 might also receive the following errors:

- 401 Unauthorized - If invalid credentials are provided.
- 403 Forbidden - If the user cannot access a specific resource or perform the requested operation.

# API versions

Lists information for all Networking API versions.

GET

/

## List API versions

Lists information about all Networking API versions.

Normal response codes: 200

Request

Response Parameters

Name	In	Type	Description
versions	body	array	List of versions.
status	body	string	Status of the API, which can be CURRENT, STABLE or DEPRECATED.
id	body	string	Version of the API.
links	body	array	List of version links. Each link is a dict with 'href' and 'rel'.
href	body	string	Link to the API.
rel	body	string	Relationship of link with the version.

Response Example

```
{
  "versions": [
    {
      "status": "CURRENT",
      "id": "v2.0",
      "links": [
        {
          "href": "http://23.253.228.211:9696/v2.0",
          "rel": "self"
        }
      ]
    }
  ]
}
```

**GET**  
/v2.0/

## Show API v2 details

Shows details for Networking API v2.0.

Normal response codes: 200

Error response codes: 401

Request

Response Parameters

Name	In	Type	Description
resources	body	array	List of resource objects.
name	body	string	Name of the resource.
collection	body	string	Collection name of the resource.
links	body	array	List of links related to the resource. Each link is a dict with 'href' and 'rel'.
href	body	string	Link to the resource.
rel	body	string	Relationship between link and the resource.

Response Example

```
{
  "resources": [
    {
      "links": [
        {
          "href": "http://23.253.228.211:9696/v2.0/subnets",
          "rel": "self"
        }
      ],
      "name": "subnet",
      "collection": "subnets"
    },
    {
      "links": [
        {
          "href": "http://23.253.228.211:9696/v2.0/networks",
          "rel": "self"
        }
      ],
      "name": "network",
      "collection": "networks"
    },
    {
      "links": [
        {
          "href": "http://23.253.228.211:9696/v2.0/ports",
          "rel": "self"
        }
      ],
      "name": "port",
      "collection": "ports"
    }
  ]
}
```

# Extensions

Extensions introduce features and vendor-specific functionality to the API.

**GET**

/v2.0/extensions

## List extensions

Lists available extensions.

Lists available Networking API v2.0 extensions and shows details for an extension.

Normal response codes: 200

Error response codes: 401

Request

Response Parameters

Name	In	Type	Description
extensions	body	array	A list of extension objects.
name	body	string	Human-readable name of the resource.
links	body	array	List of links related to the extension.
alias	body	string	The alias for the extension. For example “quotas” or “security-group”.
updated	body	string	The date and timestamp when the extension was last updated.
description	body	string	The human-readable description for the resource.

## Response Example

```
{
  "extensions": [
    {
      "updated": "2013-01-20T00:00:00-00:00",
      "name": "Neutron Service Type Management",
      "links": [],
      "alias": "service-type",
      "description": "API for retrieving service providers for Neutron advanced services"
    },
    {
      "updated": "2012-10-05T10:00:00-00:00",
      "name": "security-group",
      "links": [],
      "alias": "security-group",
      "description": "The security groups extension."
    },
    {
      "updated": "2013-02-07T10:00:00-00:00",
      "name": "L3 Agent Scheduler",
      "links": [],
      "alias": "l3_agent_scheduler",
      "description": "Schedule routers among l3 agents"
    }
  ]
}
```

```

{
  "updated": "2013-02-07T10:00:00-00:00",
  "name": "Loadbalancer Agent Scheduler",
  "links": [],
  "alias": "lbaas_agent_scheduler",
  "description": "Schedule pools among lbaas agents"
},
{
  "updated": "2013-03-28T10:00:00-00:00",
  "name": "Neutron L3 Configurable external gateway mode",
  "links": [],
  "alias": "ext-gw-mode",
  "description": "Extension of the router abstraction for specifying whether SNAT should occur on the external
gateway"
},
{
  "updated": "2014-02-03T10:00:00-00:00",
  "name": "Port Binding",
  "links": [],
  "alias": "binding",
  "description": "Expose port bindings of a virtual port to external application"
},
{
  "updated": "2012-09-07T10:00:00-00:00",
  "name": "Provider Network",
  "links": [],
  "alias": "provider",
  "description": "Expose mapping of virtual networks to physical networks"
},
{
  "updated": "2013-02-03T10:00:00-00:00",
  "name": "agent",
  "links": [],
  "alias": "agent",
  "description": "The agent management extension."
},
{
  "updated": "2012-07-29T10:00:00-00:00",
  "name": "Quota management support",
  "links": [],
  "alias": "quotas",
  "description": "Expose functions for quotas management per tenant"
},
{
  "updated": "2013-02-07T10:00:00-00:00",
  "name": "DHCP Agent Scheduler",
  "links": [],
  "alias": "dhcp_agent_scheduler",
  "description": "Schedule networks among dhcp agents"
},
{
  "updated": "2013-06-27T10:00:00-00:00",
  "name": "Multi Provider Network",
  "links": [],
  "alias": "multi-provider",
  "description": "Expose mapping of virtual networks to multiple physical networks"
},
{
  "updated": "2013-01-14T10:00:00-00:00",
  "name": "Neutron external network",
  "links": [],
  "alias": "external-net",
  "description": "Adds external network attribute to network resource."
},
{
  "updated": "2012-07-20T10:00:00-00:00",
  "name": "Neutron L3 Router",
  "links": [],
  "alias": "router",
  "description": "Router abstraction for basic L3 forwarding between L2 Neutron networks and access to external
networks via a NAT gateway."
},
{
  "updated": "2013-07-23T10:00:00-00:00",
  "name": "Allowed Address Pairs",
  "links": [],
  "alias": "allowed-address-pairs",
  "description": "Provides allowed address pairs"
},
{
  "updated": "2013-03-17T12:00:00-00:00",
  "name": "Neutron Extra DHCP opts",
  "links": [],
  "alias": "extra_dhcp_opt",
  "description": "Extra options configuration for DHCP. For example PXE boot options to DHCP clients can be
specified (e.g. tftp-server, server-ip-address, bootfile-name)"
},
{
  "updated": "2012-10-07T10:00:00-00:00",
  "name": "LoadBalancing service",
  "links": [],
  "alias": "lbaas",
  "description": "Extension for LoadBalancing service"
},
{
  "updated": "2013-02-01T10:00:00-00:00",
  "name": "Neutron Extra Route",
  "links": [],

```



```

    "alias": "extraroute",
    "description": "Extra routes configuration for L3 router"
  }
}

```

**GET**

/v2.0/extensions/{alias}

## Show extension details

Shows details for an extension, by alias. The response shows the extension name and its alias. To show details for an extension, you specify the alias.

Normal response codes: 200

Error response codes: 401, 404

### Request

Name	In	Type	Description
alias	path	string	The alias of an extension.

### Response Parameters

Name	In	Type	Description
extension	body	object	An extension object.
name	body	string	Human-readable name of the resource.
links	body	array	List of links related to the extension.
alias	body	string	The alias for the extension. For example “quotas” or “security-group”.
updated	body	string	The date and timestamp when the extension was last updated.
description	body	string	The human-readable description for the resource.

### Response Example

```

{
  "extension": {
    "updated": "2013-02-03T10:00:00-00:00",
    "name": "agent",
    "links": [],
    "alias": "agent",
    "description": "The agent management extension."
  }
}

```

# Layer 2 Networking

## Networks

Lists, shows details for, creates, updates, and deletes networks.

### provider extended attributes

The provider extension allows administrative users to define a physical binding of a logical network. This extension provides three additional attributes:

`provider:network_type`, `provider:physical_network` and `provider:segmentation_id`. The validation rules for these attributes vary across `provider:network_type`. For example, `vlan` and `flat` network types require `provider:physical_network` attribute, but `vxlan` network type does not.

Most Networking plug-ins and drivers do not support updating any provider related attributes.

### multiple provider extension

The `multi-provider` extension allows administrative users to define multiple physical bindings for a logical network.

To define multiple physical bindings for a network, include a `segments` list in the request body of network creation request. Each element in the `segments` list has the same structure as the provider network attributes. These attributes are `provider:network_type`, `provider:physical_network`, and `provider:segmentation_id`. The same validation rules are applied to each element in the `segments` list.

Note that you cannot use the provider extension and the multiple provider extension for a single logical network.

### VLAN transparency extension

The `vlan-transparent` extension enables plug-ins that support VLAN transparency to deliver VLAN transparent trunk networks. This extension introduces `vlan_transparent` attribute to control a VLAN transparency of the network. If the service does not support VLAN transparency and a user requests a VLAN transparent network, the plug-in refuses to create one and returns an appropriate error to the user.

#### GET

`/v2.0/networks/{network_id}`

### Show network details

Shows details for a network.

Use the `fields` query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
network_id	path	string	The ID of the network.
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
network	body	object	A network object.
admin_state_up	body	boolean	The administrative state of the network, which is up ( <code>true</code> ) or down ( <code>false</code> ).
availability_zone_hints	body	array	The availability zone candidate for the network.
availability_zones	body	array	The availability zone for the network.
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
id	body	string	The ID of the network.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
name	body	string	Human-readable name of the network.
port_security_enabled	body	boolean	The port security status of the network. Valid values are <code>enabled</code> ( <code>true</code> ) and <code>disabled</code> ( <code>false</code> ). This value is used as the default value of <code>port_security_enabled</code> field of a newly created port.
project_id	body	string	The ID of the project.
provider:network_type	body	string	The type of physical network that this network is mapped to. For example, <code>flat</code> , <code>vlan</code> , <code>vxlan</code> , or <code>gre</code> . Valid values depend on

Name	In	Type	Description
			a networking back-end.
provider:physical_network	body	string	The physical network where this network is implemented.
provider:segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
qos_policy_id	body	string	The ID of the QoS policy.
router:external	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments	body	array	A list of provider segment objects.
shared	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
status	body	string	The network status. Values are ACTIVE, DOWN, BUILD or ERROR.
subnets	body	array	The associated subnets.
tenant_id	body	string	The ID of the project.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).
vlan_transparent	body	boolean	Indicates the VLAN transparency mode of the network, which is VLAN transparent (true) or not VLAN transparent (false).
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "network": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "created_at": "2016-03-08T20:19:41",
    "id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
    "mtu": 0,
    "name": "private-network",
    "port_security_enabled": true,
    "project_id": "4fd44f30292945e481c7b8a0c8908869",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
```

```

    "shared": true,
    "status": "ACTIVE",
    "subnets": [
        "54d6f61d-db07-451c-9ab3-b9609b6b6f0b"
    ],
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
}
}

```

### Response Example (admin user; single segment mapping)

```

{
  "network": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
        "nova"
    ],
    "created_at": "2016-03-08T20:19:41",
    "id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
    "mtu": 0,
    "name": "private-network",
    "port_security_enabled": true,
    "project_id": "4fd44f30292945e481c7b8a0c8908869",
    "provider:network_type": "local",
    "provider:physical_network": null,
    "provider:segmentation_id": null,
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
    "shared": true,
    "status": "ACTIVE",
    "subnets": [
        "54d6f61d-db07-451c-9ab3-b9609b6b6f0b"
    ],
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
  }
}

```

### Response Example (admin user; multiple segment mappings)

```

{
  "network": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
        "nova"
    ],
    "created_at": "2016-03-08T20:19:41",
    "id": "4e8e5957-649f-477b-9e5b-f1f75b21c03c",
    "mtu": 0,
    "name": "net1",
    "port_security_enabled": true,
    "project_id": "9bacb3c5d39d41a79512987f338cf177",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
    "segments": [
        {
            "provider:network_type": "vlan",
            "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
            "provider:segmentation_id": 2
        },
        {

```

```

        "provider:network_type": "stt",
        "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
        "provider:segmentation_id": 0
    },
    "shared": false,
    "status": "ACTIVE",
    "subnets": [
        "54d6f61d-db07-451c-9ab3-b9609b6b6f0b"
    ],
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
}
}

```

**PUT**

/v2.0/networks/{network\_id}

## Update network

Updates a network.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
network_id	path	string	The ID of the network.
network	body	object	A network object.
admin_state_up (Optional)	body	boolean	The administrative state of the network, which is up (true) or down (false).
name (Optional)	body	string	Human-readable name of the network.
port_security_enabled (Optional)	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.
qos_policy_id (Optional)	body	string	The ID of the QoS policy.
router:external (Optional)	body	boolean	Indicates whether this network can provide floating IPs via a router.
shared (Optional)	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "network": {
    "name": "sample_network_5_updated",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e"
  }
}
```

## Response Parameters

Name	In	Type	Description
network	body	object	A network object.
admin_state_up	body	boolean	The administrative state of the network, which is up (true) or down (false).
availability_zone_hints	body	array	The availability zone candidate for the network.
availability_zones	body	array	The availability zone for the network.
id	body	string	The ID of the network.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
name	body	string	Human-readable name of the network.
port_security_enabled	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.
project_id	body	string	The ID of the project.
provider:network_type	body	string	The type of physical network that this network is mapped to. For example, flat, vlan, vxlan, or gre. Valid values depend on a networking back-end.
provider:physical_network	body	string	The physical network where this network is implemented.
provider:segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
qos_policy_id	body	string	The ID of the QoS policy.

Name	In	Type	Description
router:external	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments	body	array	A list of provider segment objects.
shared	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
status	body	string	The network status. Values are ACTIVE, DOWN, BUILD or ERROR.
subnets	body	array	The associated subnets.
tenant_id	body	string	The ID of the project.
description	body	string	A human-readable description for the resource.

### Response Example

This is an example when a regular user without administrative roles sends a PUT request. Response examples for administrative users are similar to responses of [Show network details](#) and [Create network](#). See them for details.

```
{
  "network": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "created_at": "2016-03-08T20:19:41",
    "id": "1f370095-98f6-4079-be64-6d3d4a6adcc6",
    "mtu": 0,
    "name": "sample_network_5_updated",
    "port_security_enabled": true,
    "project_id": "4fd44f30292945e481c7b8a0c8908869",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
    "shared": false,
    "status": "ACTIVE",
    "subnets": [
      "54d6f61d-db07-451c-9ab3-b9609b6b6f0b"
    ],
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
  }
}
```

### DELETE

/v2.0/networks/{network\_id}

### Delete network

Deletes a network and its associated resources.



Normal response codes: 204

Error response codes: 401, 404, 409

#### Request

Name	In	Type	Description
network_id	path	string	The ID of the network.

#### Response

There is no body content for the response of a successful DELETE request.

#### GET

/v2.0/networks

### List networks

Lists networks to which the project has access.

Default policy settings return only networks that the project who submits the request owns, unless an administrative user submits the request. In addition, networks shared with the project who submits the request are also returned.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

You can also use the `tags`, `tags-any`, `not-tags`, `not-tags-any` query parameter to filter the response with tags. For information, see [REST API Impact](#).

Normal response codes: 200

Error response codes: 401

#### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
networks	body	array	A list of network objects.
admin_state_up	body	boolean	The administrative state of the network, which is up (true) or down (false).
availability_zone_hints	body	array	The availability zone candidate for the network.
availability_zones	body	array	The availability zone for the network.
id	body	string	The ID of the network.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
name	body	string	Human-readable name of the network.
port_security_enabled	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.
project_id	body	string	The ID of the project.
provider:network_type	body	string	The type of physical network that this network is mapped to. For example, flat, vlan, vxlan, or gre. Valid values depend on a networking back-end.
provider:physical_network	body	string	The physical network where this network is implemented.
provider:segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
qos_policy_id	body	string	The ID of the QoS policy.
router:external	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments	body	array	A list of provider segment objects.

Name	In	Type	Description
shared	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
status	body	string	The network status. Values are ACTIVE, DOWN, BUILD or ERROR.
subnets	body	array	The associated subnets.
tenant_id	body	string	The ID of the project.
vlan_transparent	body	boolean	Indicates the VLAN transparency mode of the network, which is VLAN transparent (true) or not VLAN transparent (false).
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "networks": [
    {
      "admin_state_up": true,
      "availability_zone_hints": [],
      "availability_zones": [
        "nova"
      ],
      "created_at": "2016-03-08T20:19:41",
      "id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
      "mtu": 0,
      "name": "net1",
      "port_security_enabled": true,
      "project_id": "4fd44f30292945e481c7b8a0c8908869",
      "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
      "router:external": false,
      "shared": false,
      "status": "ACTIVE",
      "subnets": [
        "54d6f61d-db07-451c-9ab3-b9609b6b6f0b"
      ],
      "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
      "updated_at": "2016-03-08T20:19:41",
      "vlan_transparent": true,
      "description": ""
    },
    {
      "admin_state_up": true,
      "availability_zone_hints": [],
      "availability_zones": [
        "nova"
      ],
      "id": "db193ab3-96e3-4cb3-8fc5-05f4296d0324",
      "mtu": 0,
      "name": "net2",
      "port_security_enabled": true,
      "project_id": "26a7980765d0414dbc1fc1f88cdb7e6e",

```

```

    "qos_policy_id": "bfdb6c39f71e4d44b1dfbda245c50819",
    "router:external": false,
    "shared": false,
    "status": "ACTIVE",
    "subnets": [
        "08eae331-0402-425a-923c-34f7cfe39c1b"
    ],
    "tenant_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
}
]
}

```

## Response Example (admin user)

When Administrative users request to list networks, physical segment information bound to the networks are also returned in a response. In this example, a network net1 is mapped to a single network segment and a network net2 is mapped to multiple network segments.

```

{
  "networks": [
    {
      "admin_state_up": true,
      "availability_zone_hints": [],
      "availability_zones": [
        "nova"
      ],
      "created_at": "2016-03-08T20:19:41",
      "id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
      "mtu": 0,
      "name": "net1",
      "port_security_enabled": true,
      "project_id": "4fd44f30292945e481c7b8a0c8908869",
      "provider:network_type": "vlan",
      "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
      "provider:segmentation_id": 3,
      "router:external": false,
      "shared": false,
      "status": "ACTIVE",
      "subnets": [
        "54d6f61d-db07-451c-9ab3-b9609b6b6f0b"
      ],
      "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
      "updated_at": "2016-03-08T20:19:41",
      "vlan_transparent": true,
      "description": ""
    },
    {
      "admin_state_up": true,
      "availability_zone_hints": [],
      "availability_zones": [
        "nova"
      ],
      "id": "db193ab3-96e3-4cb3-8fc5-05f4296d0324",
      "mtu": 0,
      "name": "net2",
      "port_security_enabled": true,
      "project_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
      "provider:network_type": "local",
      "provider:physical_network": null,
      "provider:segmentation_id": null,
      "qos_policy_id": "bfdb6c39f71e4d44b1dfbda245c50819",
      "router:external": false,
      "segments": [

```

```

    {
      "provider:network_type": "vlan",
      "provider:physical_network": "8bab8453-1bc9-45af-8c70-
f83aa9b50453",
      "provider:segmentation_id": 2
    },
    {
      "provider:network_type": "stt",
      "provider:physical_network": "8bab8453-1bc9-45af-8c70-
f83aa9b50453",
      "provider:segmentation_id": 0
    }
  ],
  "shared": false,
  "status": "ACTIVE",
  "subnets": [
    "08eae331-0402-425a-923c-34f7cfe39c1b"
  ],
  "tenant_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
  "updated_at": "2016-03-08T20:19:41",
  "vlan_transparent": false,
  "description": ""
}
]
}

```

**POST**

/v2.0/networks

## Create network

Creates a network.

A request body is optional. An administrative user can specify another project ID, which is the project that owns the network, in the request body.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
network	body	object	A network object.
admin_state_up (Optional)	body	boolean	The administrative state of the network, which is up (true) or down (false).
name (Optional)	body	string	Human-readable name of the network.
port_security_enabled (Optional)	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.

Name	In	Type	Description
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
provider:network_type (Optional)	body	string	The type of physical network that this network should be mapped to. For example, flat, vlan, vxlan, or gre. Valid values depend on a networking back-end.
provider:physical_network (Optional)	body	string	The physical network where this network should be implemented. The Networking API v2.0 does not provide a way to list available physical networks. For example, the Open vSwitch plug-in configuration file defines a symbolic name that maps to specific bridges on each compute host.
provider:segmentation_id (Optional)	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
qos_policy_id (Optional)	body	string	The ID of the QoS policy.
router:external (Optional)	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments (Optional)	body	array	A list of provider segment objects.
shared (Optional)	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through

Name	In	Type	Description
			authorization policies.
vlan_transparent (Optional)	body	boolean	Indicates the VLAN transparency mode of the network, which is VLAN transparent (true) or not VLAN transparent (false).
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

### Request Example

```
{
  "network": {
    "name": "sample_network",
    "admin_state_up": true,
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e"
  }
}
```

### Request Example (admin user; single segment mapping)

```
{
  "network": {
    "admin_state_up": true,
    "name": "net1",
    "provider:network_type": "vlan",
    "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
    "provider:segmentation_id": 2,
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e"
  }
}
```

### Request Example (admin user; multiple segment mappings)

```
{
  "network": {
    "segments": [
      {
        "provider:segmentation_id": "2",
        "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
        "provider:network_type": "vlan"
      },
      {
        "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
        "provider:network_type": "stt"
      }
    ],
    "name": "net1",
    "admin_state_up": true,
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e"
  }
}
```

## Response Parameters

Name	In	Type	Description
network	body	object	A network object.
admin_state_up	body	boolean	The administrative state of the network, which is up (true) or down (false).
availability_zone_hints	body	array	The availability zone candidate for the network.
availability_zones	body	array	The availability zone for the network.
id	body	string	The ID of the network.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
name	body	string	Human-readable name of the network.
port_security_enabled	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.
project_id	body	string	The ID of the project.
provider:network_type	body	string	The type of physical network that this network is mapped to. For example, flat, vlan, vxlan, or gre. Valid values depend on a networking back-end.
provider:physical_network	body	string	The physical network where this network is implemented.
provider:segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.



Name	In	Type	Description
qos_policy_id	body	string	The ID of the QoS policy.
router:external	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments	body	array	A list of provider segment objects.
shared	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
status	body	string	The network status. Values are ACTIVE, DOWN, BUILD or ERROR.
subnets	body	array	The associated subnets.
tenant_id	body	string	The ID of the project.
vlan_transparent	body	boolean	Indicates the VLAN transparency mode of the network, which is VLAN transparent (true) or not VLAN transparent (false).
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "network": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "created_at": "2016-03-08T20:19:41",
    "id": "4e8e5957-649f-477b-9e5b-f1f75b21c03c",
    "mtu": 0,
    "name": "net1",
    "port_security_enabled": true,
    "project_id": "9bacb3c5d39d41a79512987f338cf177",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
    "shared": false,
    "status": "ACTIVE",
    "subnets": [],
    "tenant_id": "9bacb3c5d39d41a79512987f338cf177",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
  }
}
```

## Response Example (admin user; single segment mapping)

```
{
  "network": {
    "status": "ACTIVE",
    "subnets": [],
    "availability_zone_hints": [],
    "name": "net1",
    "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
    "admin_state_up": true,
    "project_id": "9bacb3c5d39d41a79512987f338cf177",
    "tenant_id": "9bacb3c5d39d41a79512987f338cf177",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
    "provider:network_type": "vlan",
    "mtu": 0,
    "shared": false,
    "id": "4e8e5957-649f-477b-9e5b-f1f75b21c03c",
    "provider:segmentation_id": 2,
    "description": ""
  }
}
```

## Response Example (admin user; multiple segment mappings)

```
{
  "network": {
    "status": "ACTIVE",
    "subnets": [],
    "name": "net1",
    "admin_state_up": true,
    "project_id": "9bacb3c5d39d41a79512987f338cf177",
    "tenant_id": "9bacb3c5d39d41a79512987f338cf177",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "segments": [
      {
        "provider:segmentation_id": 2,
        "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
        "provider:network_type": "vlan"
      }
    ],
  }
}
```

```

    {
      "provider:segmentation_id": null,
      "provider:physical_network": "8bab8453-1bc9-45af-8c70-f83aa9b50453",
      "provider:network_type": "stt"
    },
    "shared": false,
    "id": "4e8e5957-649f-477b-9e5b-f1f75b21c03c",
    "description": ""
  }
}

```

**POST**

/v2.0/networks

## Bulk create networks

Creates multiple networks in a single request.

In the request body, specify a list of networks.

The bulk create operation is always atomic. Either all or no networks in the request body are created.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
networks	body	array	A list of network objects.
admin_state_up (Optional)	body	boolean	The administrative state of the network, which is up (true) or down (false).
name (Optional)	body	string	Human-readable name of the network.
port_security_enabled (Optional)	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
provider:network_type (Optional)	body	string	The type of physical network that this network should be mapped to. For example, flat, vlan, vxlan, or gre. Valid values depend on a networking back-end.
provider:physical_network (Optional)	body	string	The physical network where this network should be implemented. The

Name	In	Type	Description
			Networking API v2.0 does not provide a way to list available physical networks. For example, the Open vSwitch plug-in configuration file defines a symbolic name that maps to specific bridges on each compute host.
provider:segmentation_id (Optional)	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
qos_policy_id (Optional)	body	string	The ID of the QoS policy.
router:external (Optional)	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments (Optional)	body	array	A list of provider segment objects.
shared (Optional)	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
vlan_transparent (Optional)	body	boolean	Indicates the VLAN transparency mode of the network, which is VLAN transparent (true) or not VLAN transparent (false).
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "networks": [
    {
      "admin_state_up": true,
      "name": "sample_network3",
      "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e"
    },
    {
      "admin_state_up": true,
      "name": "sample_network4",
      "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e"
    }
  ]
}
```

## Response Parameters

Name	In	Type	Description
networks	body	array	A list of network objects.
admin_state_up	body	boolean	The administrative state of the network, which is up (true) or down (false).
availability_zone_hints	body	array	The availability zone candidate for the network.
availability_zones	body	array	The availability zone for the network.
id	body	string	The ID of the network.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
name	body	string	Human-readable name of the network.
port_security_enabled	body	boolean	The port security status of the network. Valid values are enabled (true) and disabled (false). This value is used as the default value of port_security_enabled field of a newly created port.
project_id	body	string	The ID of the project.
provider:network_type	body	string	The type of physical network that this network is mapped to. For example, flat, vlan, vxlan, or gre. Valid values depend on a networking back-end.
provider:physical_network	body	string	The physical network where this network is implemented.
provider:segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
qos_policy_id	body	string	The ID of the QoS policy.

Name	In	Type	Description
router:external	body	boolean	Indicates whether this network can provide floating IPs via a router.
segments	body	array	A list of provider segment objects.
shared	body	boolean	Indicates whether this network is shared across all tenants. By default, only administrative users can change this value.
status	body	string	The network status. Values are ACTIVE, DOWN, BUILD or ERROR.
subnets	body	array	The associated subnets.
tenant_id	body	string	The ID of the project.
vlan_transparent	body	boolean	Indicates the VLAN transparency mode of the network, which is VLAN transparent (true) or not VLAN transparent (false).
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "networks": [
    {
      "admin_state_up": true,
      "availability_zone_hints": [],
      "availability_zones": [
        "nova"
      ],
      "created_at": "2016-03-08T20:19:41",
      "id": "bc1a76cb-8767-4c3a-bb95-018b822f2130",
      "mtu": 0,
      "name": "sample_network3",
      "project_id": "4fd44f30292945e481c7b8a0c8908869",
      "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
      "router:external": false,
      "shared": false,
      "status": "ACTIVE",
      "subnets": [],
      "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
      "updated_at": "2016-03-08T20:19:41",
      "vlan_transparent": false,
      "description": ""
    },
    {
      "admin_state_up": true,
      "availability_zone_hints": [],
      "availability_zones": [
        "nova"
      ],
      "created_at": "2016-03-08T20:19:41",
      "id": "bc1a76cb-8767-4c3a-bb95-018b822f2130",
      "mtu": 0,
      "name": "sample_network3",
      "project_id": "4fd44f30292945e481c7b8a0c8908869",
      "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
      "router:external": false,
      "shared": false,
      "status": "ACTIVE",
      "subnets": [],
      "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
      "updated_at": "2016-03-08T20:19:41",
      "vlan_transparent": false,
      "description": ""
    }
  ]
}
```

```

    "created_at": "2016-03-08T20:19:41",
    "id": "af374017-c9ae-4a1d-b799-ab73111476e2",
    "mtu": 0,
    "name": "sample_network4",
    "project_id": "4fd44f30292945e481c7b8a0c8908869",
    "qos_policy_id": "6a8454ade84346f59e8d40665f878b2e",
    "router:external": false,
    "shared": false,
    "status": "ACTIVE",
    "subnets": [],
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "updated_at": "2016-03-08T20:19:41",
    "vlan_transparent": false,
    "description": ""
  }
}

```

## Ports

Lists, shows details for, creates, updates, and deletes ports.

### Port binding extended attributes

The port binding extension (binding) allows administrative users to specify and retrieve physical binding information of ports. The extension defines several attributes whose names have a prefix binding: including binding:host\_id, binding:vnic\_type, binding:vif\_type, binding:vif\_details, and binding:profile.

**GET**

/v2.0/ports/{port\_id}

### Show port details

Shows details for a port.

Use the fields query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 404

### Request

Name	In	Type	Description
port_id	path	string	The ID of the port.

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using fields parameter, the API returns only the requested set of attributes. fields parameter can be specified multiple times. For example, if you specify fields=id&fields=name in the request URL, only id and name attributes will be returned.

## Response Parameters

Name	In	Type	Description
port	body	object	A port object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
allowed_address_pairs	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs.
binding:host_id	body	string	The ID of the host where the port resides.
binding:profile	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field.
binding:vif_details	body	object	A dictionary which contains additional information on the port. Currently the following fields are defined: port_filter and ovs_hybrid_plug. port_filter is a boolean indicating the networking service provides port filtering features such as security group and/or anti MAC/IP spoofing. ovs_hybrid_plug is a boolean used to inform an API consumer like nova that the hybrid plugging strategy for OVS should be used.
binding:vif_type	body	string	The type of which mechanism is used for the port. An API consumer like nova can use this to determine an appropriate way to attach a device (for example an interface of a virtual server) to the port. Available values currently defined includes ovs, bridge, macvtap, hw_veb, hostdev_physical, vhostuser, distributed and other. There are



Name	In	Type	Description
			also special values: unbound and binding_failed. unbound means the port is not bound to a networking back-end. binding_failed means an error that the port failed to be bound to a networking back-end.
binding:vnic_type	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments.
created_at	body	string	Time at which port has been created.
description	body	string	A human-readable description for the resource.
device_id	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips	body	array	The IP addresses for the port. If the port has multiple IP addresses, this field has multiple entries. Each entry consists of IP address (ip_address) and the subnet ID from which the IP address is assigned (subnet_id).
id	body	string	The ID of the resource.
mac_address	body	string	The MAC address of the port.
name	body	string	Human-readable name of the resource.
network_id	body	string	The ID of the attached network.
port_security_enabled	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id	body	string	The ID of the project.
security_groups	body	array	The IDs of security groups applied to

Name	In	Type	Description
			the port.
status	body	string	The port status. Values are ACTIVE, DOWN, BUILD and ERROR.
tenant_id	body	string	The ID of the project.
updated_at	body	string	Time at which port has been updated.

## Response Example

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "created_at": "2016-03-08T20:19:41",
    "description": "",
    "device_id": "5e3898d7-11be-483e-9732-b2f5eccd2b2e",
    "device_owner": "network:router_interface",
    "extra_dhcp_opts": [],
    "fixed_ips": [
      {
        "ip_address": "10.0.0.1",
        "subnet_id": "a0304c3a-4f08-4c43-88af-d796509c97d2"
      }
    ],
    "id": "46d4bfb9-b26e-41f3-bd2e-e6dcc1ccedb2",
    "mac_address": "fa:16:3e:23:fd:d7",
    "name": "",
    "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
    "port_security_enabled": false,
    "project_id": "7e02058126cc4950b75f9970368ba177",
    "security_groups": [],
    "status": "ACTIVE",
    "tenant_id": "7e02058126cc4950b75f9970368ba177",
    "updated_at": "2016-03-08T20:19:41"
  }
}
```

## Response Example (admin user)

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "devstack",
    "binding:profile": {},
    "binding:vif_details": {
      "ovs_hybrid_plug": true,
      "port_filter": true
    },
    "binding:vif_type": "ovs",
    "binding:vnictype": "normal",
    "created_at": "2016-03-08T20:19:41",
    "description": "",
    "device_id": "5e3898d7-11be-483e-9732-b2f5eccd2b2e",
    "device_owner": "network:router_interface",
    "extra_dhcp_opts": [],
    "fixed_ips": [
      {
        "ip_address": "10.0.0.1",
        "subnet_id": "a0304c3a-4f08-4c43-88af-d796509c97d2"
      }
    ],
    "id": "46d4bfb9-b26e-41f3-bd2e-e6dcc1ccedb2",
    "mac_address": "fa:16:3e:23:fd:d7",
    "name": "",
    "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
    "port_security_enabled": false,
    "project_id": "7e02058126cc4950b75f9970368ba177",
    "security_groups": [],
    "status": "ACTIVE",
    "tenant_id": "7e02058126cc4950b75f9970368ba177",
    "updated_at": "2016-03-08T20:19:41"
  }
}
```

```

    "id": "46d4bfb9-b26e-41f3-bd2e-e6dcc1ccedb2",
    "mac_address": "fa:16:3e:23:fd:d7",
    "name": "",
    "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
    "port_security_enabled": false,
    "project_id": "7e02058126cc4950b75f9970368ba177",
    "security_groups": [],
    "status": "ACTIVE",
    "tenant_id": "7e02058126cc4950b75f9970368ba177",
    "updated_at": "2016-03-08T20:19:41"
  }
}

```

**PUT**

/v2.0/ports/{port\_id}

## Update port

Updates a port.

You can update information for a port, such as its symbolic name and associated IPs. When you update IPs for a port, any previously associated IPs are removed, returned to the respective subnet allocation pools, and replaced by the IPs in the request body. Therefore, this operation replaces the `fixed_ip` attribute when you specify it in the request body. If the updated IP addresses are not valid or are already in use, the operation fails and the existing IP addresses are not removed from the port.

When you update security groups for a port and the operation succeeds, any associated security groups are removed and replaced by the security groups in the request body. Therefore, this operation replaces the `security_groups` attribute when you specify it in the request body. If the security groups are not valid, the operation fails and the existing security groups are not removed from the port.

Normal response codes: 200

Error response codes: 400, 401, 403, 404, 409

## Request

Name	In	Type	Description
port_id	path	string	The ID of the port.
port	body	object	A port object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
allowed_address_pairs (Optional)	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs. The default is an empty list. For each address pair, ip_address is required and IP address or CIDR can be specified. mac_address is optional and if unspecified the MAC

Name	In	Type	Description
			address of the port is used as default.
binding:host_id (Optional)	body	string	The ID of the host where the port resides. The default is an empty string.
binding:profile (Optional)	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field. The default is an empty dictionary.
binding:vnic_type (Optional)	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments. The default is normal.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
device_id (Optional)	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner (Optional)	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts (Optional)	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips (Optional)	body	array	<p>The IP addresses for the port. If you would like to assign multiple IP addresses for the port, specify multiple entries in this field. Each entry consists of IP address (ip_address) and the subnet ID from which the IP address is assigned (subnet_id).</p> <ul style="list-style-type: none"> <li>• If you specify both a subnet ID and an IP address, OpenStack Networking tries to allocate the IP address on that subnet to the port.</li> <li>• If you specify only a subnet ID, OpenStack Networking allocates an available IP from that subnet to the port.</li> <li>• If you specify only an IP address, OpenStack Networking tries to allocate the IP address if the address is a valid IP for any of the subnets on the specified network.</li> </ul>
mac_address (Optional)	body	string	The MAC address of the port. By default, only administrative users can change this value.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
port_security_enabled (Optional)	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.

Name	In	Type	Description
security_groups (Optional)	body	array	The IDs of security groups applied to the port.

## Request Example

```
{
  "port": {
    "admin_state_up": true,
    "device_id": "d90a13da-be41-461f-9f99-1dbcf438fdf2",
    "device_owner": "compute:nova",
    "name": "test-for-port-update"
  }
}
```

## Request Example (admin user)

```
{
  "port": {
    "binding:host_id": "test_for_port_update_host",
    "device_id": "d90a13da-be41-461f-9f99-1dbcf438fdf2",
    "device_owner": "compute:nova"
  }
}
```

## Response Parameters

Name	In	Type	Description
port	body	object	A port object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
allowed_address_pairs	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs.
binding:host_id	body	string	The ID of the host where the port resides.
binding:profile	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field.
binding:vif_details	body	object	A dictionary which contains additional information on the port. Currently the following fields are defined: port_filter and ovs_hybrid_plug. port_filter is a boolean indicating the networking service provides port filtering features such as security

Name	In	Type	Description
			group and/or anti MAC/IP spoofing. ovs_hybrid_plug is a boolean used to inform an API consumer like nova that the hybrid plugging strategy for OVS should be used.
binding:vif_type	body	string	The type of which mechanism is used for the port. An API consumer like nova can use this to determine an appropriate way to attach a device (for example an interface of a virtual server) to the port. Available values currently defined includes ovs, bridge, macvtap, hw_veb, hostdev_physical, vhostuser, distributed and other. There are also special values: unbound and binding_failed. unbound means the port is not bound to a networking back-end. binding_failed means an error that the port failed to be bound to a networking back-end.
binding:vnic_type	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments.
created_at	body	string	Time at which port has been created.
description	body	string	A human-readable description for the resource.
device_id	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips	body	array	The IP addresses for the port. If the port has multiple IP addresses, this field has multiple entries. Each entry consists of IP address (ip_address) and the subnet ID from which the IP address is assigned (subnet_id).
id	body	string	The ID of the resource.
mac_address	body	string	The MAC address of the port.
name	body	string	Human-readable name of the

Name	In	Type	Description
			resource.
network_id	body	string	The ID of the attached network.
port_security_enabled	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id	body	string	The ID of the project.
security_groups	body	array	The IDs of security groups applied to the port.
status	body	string	The port status. Values are ACTIVE, DOWN, BUILD and ERROR.
tenant_id	body	string	The ID of the project.
updated_at	body	string	Time at which port has been updated.

## Response Example

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "test_for_port_update_host",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "binding_failed",
    "binding:vnictype": "normal",
    "description": "",
    "device_id": "d90a13da-be41-461f-9f99-1dbcf438fdf2",
    "device_owner": "compute:nova",
    "extra_dhcp_opts": [],
    "fixed_ips": [
      {
        "ip_address": "20.20.0.4",
        "subnet_id": "898dec4a-74df-4193-985f-c76721bcc746"
      }
    ],
    "id": "43c831e0-19ce-4a76-9a49-57b57e69428b",
    "mac_address": "fa:16:3e:11:11:5e",
    "name": "test-for-port-update",
    "network_id": "883fc383-5ea1-4c8b-8916-e1ddb0a9f365",
    "project_id": "522eda8d23124b25bf03fe44f1986b74",
    "security_groups": [
      "ce0179d6-8a94-4f7c-91c2-f3038e2acbd0"
    ],
    "status": "DOWN",
    "tenant_id": "522eda8d23124b25bf03fe44f1986b74"
  }
}
```

## Response Example (admin user)

```
{
  "port": {
```

```

"admin_state_up": true,
"allowed_address_pairs": [],
"binding:host_id": "test_for_port_update_host",
"binding:profile": {},
"binding:vif_details": {},
"binding:vif_type": "binding_failed",
"binding:vnic_type": "normal",
"description": "",
"device_id": "d90a13da-be41-461f-9f99-1dbcf438fdf2",
"device_owner": "compute:nova",
"extra_dhcp_opts": [],
"fixed_ips": [
  {
    "ip_address": "20.20.0.4",
    "subnet_id": "898dec4a-74df-4193-985f-c76721bcc746"
  }
],
"id": "43c831e0-19ce-4a76-9a49-57b57e69428b",
"mac_address": "fa:16:3e:11:11:5e",
"name": "test-for-port-update",
"network_id": "883fc383-5ea1-4c8b-8916-e1ddb0a9f365",
"project_id": "522eda8d23124b25bf03fe44f1986b74",
"security_groups": [
  "ce0179d6-8a94-4f7c-91c2-f3038e2acbd0"
],
"status": "DOWN",
"tenant_id": "522eda8d23124b25bf03fe44f1986b74"
}
}

```

## DELETE

/v2.0/ports/{port\_id}

### Delete port

Deletes a port.

Any IP addresses that are associated with the port are returned to the respective subnets allocation pools.

Normal response codes: 204

Error response codes: 401, 403, 404

Request

Name	In	Type	Description
port_id	path	string	The ID of the port.

Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/ports

### List ports

Lists ports to which the user has access.



Default policy settings return only those ports that are owned by the project of the user who submits the request, unless the request is submitted by a user with administrative rights.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
ports	body	array	A list of port objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
allowed_address_pairs	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of <code>{"ip_address": "&lt;IP address or CIDR&gt;", "mac_address": "&lt;MAC address&gt;"}</code> . A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs.
binding:host_id	body	string	The ID of the host where the port resides.
binding:profile	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field.
binding:vif_details	body	object	A dictionary which contains additional information on the port. Currently the following fields are defined: <code>port_filter</code> and <code>ovs_hybrid_plug</code> . <code>port_filter</code> is a boolean indicating the networking service provides port filtering features such as security group and/or anti MAC/IP spoofing. <code>ovs_hybrid_plug</code> is a boolean used to inform an API consumer like nova that the hybrid plugging strategy for OVS should be used.
binding:vif_type	body	string	The type of which mechanism is used for the port. An

Name	In	Type	Description
			API consumer like nova can use this to determine an appropriate way to attach a device (for example an interface of a virtual server) to the port. Available values currently defined includes ovs, bridge, macvtap, hw_veth, hostdev_physical, vhostuser, distributed and other. There are also special values: unbound and binding_failed. unbound means the port is not bound to a networking back-end. binding_failed means an error that the port failed to be bound to a networking back-end.
binding:vnic_type	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments.
created_at	body	string	Time at which port has been created.
description	body	string	A human-readable description for the resource.
device_id	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips	body	array	The IP addresses for the port. If the port has multiple IP addresses, this field has multiple entries. Each entry consists of IP address (ip_address) and the subnet ID from which the IP address is assigned (subnet_id).
id	body	string	The ID of the resource.
mac_address	body	string	The MAC address of the port.
name	body	string	Human-readable name of the resource.
network_id	body	string	The ID of the attached network.
port_security_enabled	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id	body	string	The ID of the project.
security_groups	body	array	The IDs of security groups applied to the port.
status	body	string	The port status. Values are ACTIVE, DOWN, BUILD and ERROR.
tenant_id	body	string	The ID of the project.

Name	In	Type	Description
updated_at	body	string	Time at which port has been updated.

## Response Example

```
{
  "ports": [
    {
      "admin_state_up": true,
      "allowed_address_pairs": [],
      "description": "",
      "device_id": "9ae135f4-b6e0-4dad-9e91-3c223e385824",
      "device_owner": "network:router_gateway",
      "extra_dhcp_opts": [],
      "fixed_ips": [
        {
          "ip_address": "172.24.4.2",
          "subnet_id": "008ba151-0b8c-4a67-98b5-0d2b87666062"
        }
      ],
      "id": "d80b1a3b-4fc1-49f3-952e-1e2ab7081d8b",
      "mac_address": "fa:16:3e:58:42:ed",
      "name": "",
      "network_id": "70c1db1f-b701-45bd-96e0-a313ee3430b3",
      "project_id": "",
      "security_groups": [],
      "status": "ACTIVE",
      "tenant_id": ""
    },
    {
      "admin_state_up": true,
      "allowed_address_pairs": [],
      "description": "",
      "device_id": "9ae135f4-b6e0-4dad-9e91-3c223e385824",
      "device_owner": "network:router_interface",
      "extra_dhcp_opts": [],
      "fixed_ips": [
        {
          "ip_address": "10.0.0.1",
          "subnet_id": "288bf4a1-51ba-43b6-9d0a-520e9005db17"
        }
      ],
      "id": "f71a6703-d6de-4be1-a91a-a570ede1d159",
      "mac_address": "fa:16:3e:bb:3c:e4",
      "name": "",
      "network_id": "f27aa545-cbdd-4907-b0c6-c9e8b039dcc2",
      "project_id": "d397de8a63f341818f198abb0966f6f3",
      "security_groups": [],
      "status": "ACTIVE",
      "tenant_id": "d397de8a63f341818f198abb0966f6f3"
    }
  ]
}
```

## Response Example (admin user)

```
{
  "ports": [
    {
      "admin_state_up": true,
      "allowed_address_pairs": [],
      "binding:host_id": "devstack",
      "binding:profile": {},
      "binding:vif_details": {
```

```

        "ovs_hybrid_plug": true,
        "port_filter": true
    },
    "binding:vif_type": "ovs",
    "binding:vnic_type": "normal",
    "description": "",
    "device_id": "9ae135f4-b6e0-4dad-9e91-3c223e385824",
    "device_owner": "network:router_gateway",
    "extra_dhcp_opts": [],
    "fixed_ips": [
        {
            "ip_address": "172.24.4.2",
            "subnet_id": "008ba151-0b8c-4a67-98b5-0d2b87666062"
        }
    ],
    "id": "d80b1a3b-4fc1-49f3-952e-1e2ab7081d8b",
    "mac_address": "fa:16:3e:58:42:ed",
    "name": "",
    "network_id": "70c1db1f-b701-45bd-96e0-a313ee3430b3",
    "port_security_enabled": true,
    "project_id": "",
    "security_groups": [],
    "status": "ACTIVE",
    "tenant_id": ""
},
{
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "devstack",
    "binding:profile": {},
    "binding:vif_details": {
        "ovs_hybrid_plug": true,
        "port_filter": true
    },
    "binding:vif_type": "ovs",
    "binding:vnic_type": "normal",
    "description": "",
    "device_id": "9ae135f4-b6e0-4dad-9e91-3c223e385824",
    "device_owner": "network:router_interface",
    "extra_dhcp_opts": [],
    "fixed_ips": [
        {
            "ip_address": "10.0.0.1",
            "subnet_id": "288bf4a1-51ba-43b6-9d0a-520e9005db17"
        }
    ],
    "id": "f71a6703-d6de-4be1-a91a-a570ede1d159",
    "mac_address": "fa:16:3e:bb:3c:e4",
    "name": "",
    "network_id": "f27aa545-cbdd-4907-b0c6-c9e8b039dcc2",
    "port_security_enabled": true,
    "project_id": "d397de8a63f341818f198abb0966f6f3",
    "security_groups": [],
    "status": "ACTIVE",
    "tenant_id": "d397de8a63f341818f198abb0966f6f3"
}
]
}

```

**POST**

/v2.0/ports

## Create port

Creates a port on a network.

To define the network in which to create the port, specify the `network_id` attribute in the request body.

Normal response codes: 201

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
port	body	object	A port object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
allowed_address_pairs (Optional)	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs. The default is an empty list. For each address pair, ip_address is required and IP address or CIDR can be specified. mac_address is optional and if unspecified the MAC address of the port is used as default.
binding:host_id (Optional)	body	string	The ID of the host where the port resides. The default is an empty string.
binding:profile (Optional)	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field. The default is an empty dictionary.
binding:vnic_type (Optional)	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments. The default is normal.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
device_id (Optional)	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner (Optional)	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts (Optional)	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips (Optional)	body	array	The IP addresses for the port. If you would like to assign multiple IP addresses for the port, specify multiple entries in this field. Each entry consists of IP address (ip_address) and the subnet ID from which

Name	In	Type	Description
			<p>the IP address is assigned (subnet_id).</p> <ul style="list-style-type: none"> <li>If you specify both a subnet ID and an IP address, OpenStack Networking tries to allocate the IP address on that subnet to the port.</li> <li>If you specify only a subnet ID, OpenStack Networking allocates an available IP from that subnet to the port.</li> <li>If you specify only an IP address, OpenStack Networking tries to allocate the IP address if the address is a valid IP for any of the subnets on the specified network.</li> </ul>
mac_address (Optional)	body	string	The MAC address of the port. If unspecified, a MAC address is automatically generated.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
network_id	body	string	The ID of the attached network.
port_security_enabled (Optional)	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
security_groups (Optional)	body	array	The IDs of security groups applied to the port.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

### Request Example

```
{
  "port": {
    "admin_state_up": true,
    "name": "private-port",
    "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7"
  }
}
```

### Request Example (admin user)

```
{
  "port": {
    "binding:host_id": "4df8d9ff-6f6f-438f-90a1-ef660d4586ad",
    "binding:profile": {
      "local_link_information": [
        {
          "port_id": "Ethernet3/1",
          "switch_id": "0a:1b:2c:3d:4e:5f",
          "switch_info": "switch1"
        }
      ]
    }
  }
}
```

```

    },
    "binding:vnic_type": "baremetal",
    "device_id": "d90a13da-be41-461f-9f99-1dbcf438fdf2",
    "device_owner": "baremetal:none"
}
}
]

```

## Response Parameters

Name	In	Type	Description
port	body	object	A port object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
allowed_address_pairs	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs.
binding:host_id	body	string	The ID of the host where the port resides.
binding:profile	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field.
binding:vif_details	body	object	A dictionary which contains additional information on the port. Currently the following fields are defined: port_filter and ovs_hybrid_plug. port_filter is a boolean indicating the networking service provides port filtering features such as security group and/or anti MAC/IP spoofing. ovs_hybrid_plug is a boolean used to inform an API consumer like nova that the hybrid plugging strategy for OVS should be used.
binding:vif_type	body	string	The type of which mechanism is used for the port. An API consumer like nova can use this to determine an appropriate way to attach a device (for example an interface of a virtual server) to the port. Available values currently defined includes ovs, bridge, macvtap, hw_veb, hostdev_physical, vhostuser, distributed and other. There are also special values: unbound and binding_failed. unbound means the port is not bound to a networking back-end. binding_failed means an

Name	In	Type	Description
			error that the port failed to be bound to a networking back-end.
binding:vnic_type	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments.
created_at	body	string	Time at which port has been created.
description	body	string	A human-readable description for the resource.
device_id	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips	body	array	The IP addresses for the port. If the port has multiple IP addresses, this field has multiple entries. Each entry consists of IP address (ip_address) and the subnet ID from which the IP address is assigned (subnet_id).
id	body	string	The ID of the resource.
mac_address	body	string	The MAC address of the port.
name	body	string	Human-readable name of the resource.
network_id	body	string	The ID of the attached network.
port_security_enabled	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id	body	string	The ID of the project.
security_groups	body	array	The IDs of security groups applied to the port.
status	body	string	The port status. Values are ACTIVE,



Name	In	Type	Description
			DOWN, BUILD and ERROR.
tenant_id	body	string	The ID of the project.
updated_at	body	string	Time at which port has been updated.

## Response Example

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "created_at": "2016-03-08T20:19:41",
    "description": "",
    "device_id": "",
    "device_owner": "",
    "extra_dhcp_opts": [],
    "fixed_ips": [
      {
        "ip_address": "10.0.0.2",
        "subnet_id": "a0304c3a-4f08-4c43-88af-d796509c97d2"
      }
    ],
    "id": "65c0ee9f-d634-4522-8954-51021b570b0d",
    "mac_address": "fa:16:3e:c9:cb:f0",
    "name": "private-port",
    "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
    "port_security_enabled": true,
    "project_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa",
    "security_groups": [
      "f0ac4394-7e4a-4409-9701-ba8be283dbc3"
    ],
    "status": "DOWN",
    "tenant_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa",
    "updated_at": "2016-03-08T20:19:41"
  }
}
```

## Response Example (admin user)

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "4df8d9ff-6f6f-438f-90a1-ef660d4586ad",
    "binding:profile": {
      "local_link_information": [
        {
          "port_id": "Ethernet3/1",
          "switch_id": "0a:1b:2c:3d:4e:5f",
          "switch_info": "switch1"
        }
      ]
    },
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnic_type": "other",
    "description": "",
    "device_id": "d90a13da-be41-461f-9f99-1dbcf438fdf2",
    "device_owner": "baremetal:none",
    "fixed_ips": [
      {
        "ip_address": "10.0.0.2",
        "subnet_id": "a0304c3a-4f08-4c43-88af-d796509c97d2"
      }
    ]
  }
}
```

```

    },
    "id": "65c0ee9f-d634-4522-8954-51021b570b0d",
    "mac_address": "fa:16:3e:c9:cb:f0",
    "name": "private-port",
    "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
    "project_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa",
    "security_groups": [
        "f0ac4394-7e4a-4409-9701-ba8be283dbc3"
    ],
    "status": "DOWN",
    "tenant_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa"
}
}

```

**POST**

/v2.0/ports

## Bulk create ports

Creates multiple ports in a single request. Specify a list of ports in the request body.

Guarantees the atomic completion of the bulk operation.

Normal response codes: 201

Error response codes: 400, 401, 403, 404, 409

## Request

Name	In	Type	Description
ports	body	array	A list of port objects.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
allowed_address_pairs (Optional)	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs. The default is an empty list. For each address pair, ip_address is required and IP address or CIDR can be specified. mac_address is optional and if unspecified the MAC address of the port is used as default.
binding:host_id (Optional)	body	string	The ID of the host where the port resides. The default is an empty string.
binding:profile (Optional)	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field. The default is an empty dictionary.
binding:vnic_type (Optional)	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments. The

Name	In	Type	Description
			default is normal.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
device_id (Optional)	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner (Optional)	body	string	The entity type that uses this port. For example, <code>compute:nova</code> (server instance), <code>network:dhcp</code> (DHCP agent) or <code>network:router_interface</code> (router interface).
extra_dhcp_opts (Optional)	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips (Optional)	body	array	<p>The IP addresses for the port. If you would like to assign multiple IP addresses for the port, specify multiple entries in this field. Each entry consists of IP address (<code>ip_address</code>) and the subnet ID from which the IP address is assigned (<code>subnet_id</code>).</p> <ul style="list-style-type: none"> <li>• If you specify both a subnet ID and an IP address, OpenStack Networking tries to allocate the IP address on that subnet to the port.</li> <li>• If you specify only a subnet ID, OpenStack Networking allocates an available IP from that subnet to the port.</li> <li>• If you specify only an IP address, OpenStack Networking tries to allocate the IP address if the address is a valid IP for any of the subnets on the specified network.</li> </ul>
mac_address (Optional)	body	string	The MAC address of the port. If unspecified, a MAC address is automatically generated.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
network_id	body	string	The ID of the attached network.
port_security_enabled (Optional)	body	boolean	The port security status. A valid value is <code>enabled</code> ( <code>true</code> ) or <code>disabled</code> ( <code>false</code> ). If port security is enabled for the port, security group rules and anti-spoofing rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
security_groups (Optional)	body	array	The IDs of security groups applied to the port.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

## Request Example

```
{
  "ports": [
    {
      "admin_state_up": false,
      "name": "sample_port_1",
      "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7"
    },
    {
      "admin_state_up": false,
      "name": "sample_port_2",
      "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7"
    }
  ]
}
```

## Response Parameters

Name	In	Type	Description
ports	body	array	A list of port objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
allowed_address_pairs	body	array	A set of zero or more allowed address pairs. An address pair consists of an IP address range and MAC address with the format of {"ip_address": "<IP address or CIDR>", "mac_address": "<MAC address>"}. A server connected to the port can send a packet with source address which matches one of the specified allowed address pairs.
binding:host_id	body	string	The ID of the host where the port resides.
binding:profile	body	string	A dictionary that enables the application running on the specific host to pass and receive vif port information specific to the networking back-end. The networking API does not define a specific format of this field.
binding:vif_details	body	object	A dictionary which contains additional information on the port. Currently the following fields are defined: port_filter and ovs_hybrid_plug. port_filter is a boolean indicating the networking service provides port filtering features such as security group and/or anti MAC/IP spoofing. ovs_hybrid_plug is a boolean used to inform an API consumer like nova that the hybrid plugging strategy for OVS should be used.
binding:vif_type	body	string	The type of which mechanism is used for the port. An API consumer like nova can use this to determine an

Name	In	Type	Description
			appropriate way to attach a device (for example an interface of a virtual server) to the port. Available values currently defined includes ovs, bridge, macvtap, hw_veb, hostdev_physical, vhostuser, distributed and other. There are also special values: unbound and binding_failed. unbound means the port is not bound to a networking back-end. binding_failed means an error that the port failed to be bound to a networking back-end.
binding:vnic_type	body	string	The type of vNIC which this port should be attached to. This is used to determine which mechanism driver(s) to be used to bind the port. The valid values are normal, macvtap, direct, baremetal, and direct-physical. What type of vNIC is actually available depends on deployments.
created_at	body	string	Time at which port has been created.
description	body	string	A human-readable description for the resource.
device_id	body	string	The ID of the device that uses this port. For example, a server instance or a logical router.
device_owner	body	string	The entity type that uses this port. For example, compute:nova (server instance), network:dhcp (DHCP agent) or network:router_interface (router interface).
extra_dhcp_opts	body	array	A set of zero or more extra DHCP option pairs. An option pair consists of an option value and name.
fixed_ips	body	array	The IP addresses for the port. If the port has multiple IP addresses, this field has multiple entries. Each entry consists of IP address (ip_address) and the subnet ID from which the IP address is assigned (subnet_id).
id	body	string	The ID of the resource.
mac_address	body	string	The MAC address of the port.
name	body	string	Human-readable name of the resource.
network_id	body	string	The ID of the attached network.
port_security_enabled	body	boolean	The port security status. A valid value is enabled (true) or disabled (false). If port security is enabled for the port,

Name	In	Type	Description
			rules are applied to the traffic on the port. If disabled, no such rules are applied.
project_id	body	string	The ID of the project.
security_groups	body	array	The IDs of security groups applied to the port.
status	body	string	The port status. Values are ACTIVE, DOWN, BUILD and ERROR.
tenant_id	body	string	The ID of the project.
updated_at	body	string	Time at which port has been updated.

## Response Example

```
{
  "ports": [
    {
      "admin_state_up": false,
      "allowed_address_pairs": [],
      "description": "",
      "device_id": "",
      "device_owner": "",
      "fixed_ips": [
        {
          "ip_address": "10.0.0.5",
          "subnet_id": "a0304c3a-4f08-4c43-88af-d796509c97d2"
        }
      ],
      "id": "94225baa-9d3f-4b93-bf12-b41e7ce49cdb",
      "mac_address": "fa:16:3e:48:b8:9f",
      "name": "sample_port_1",
      "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
      "project_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa",
      "security_groups": [
        "f0ac4394-7e4a-4409-9701-ba8be283dbc3"
      ],
      "status": "DOWN",
      "tenant_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa"
    },
    {
      "admin_state_up": false,
      "allowed_address_pairs": [],
      "description": "",
      "device_id": "",
      "device_owner": "",
      "fixed_ips": [
        {
          "ip_address": "10.0.0.6",
          "subnet_id": "a0304c3a-4f08-4c43-88af-d796509c97d2"
        }
      ],
      "id": "235b09e0-63c4-47f1-b221-66ba54c21760",
      "mac_address": "fa:16:3e:f4:73:df",
      "name": "sample_port_2",
      "network_id": "a87cc70a-3e15-4acf-8205-9b711a3531b7",
      "project_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa",
      "security_groups": [
        "f0ac4394-7e4a-4409-9701-ba8be283dbc3"
      ],
      "status": "DOWN",
    }
  ]
}
```

```
}
  ]
}
    "tenant_id": "d6700c0c9ffa4f1cb322cd4a1f3906fa"
```

## Segments

Lists, shows details for, creates, updates, and deletes segments. The segments API is admin-only.

**GET**

/v2.0/segments/{segment\_id}

### Show segment details

Shows details for a segment.

You can control which response parameters are returned by using the fields query parameter. For information, see [Filtering and column selection](#).

Normal response codes: 200

Error response codes: 401, 404

### Request

Name	In	Type	Description
segment_id	path	string	The UUID of the segment.

### Response Parameters

Name	In	Type	Description
id	body	string	The UUID of the segment.
network_id	body	string	The ID of the attached network.
physical_network	body	string	The physical network where this network is implemented.
network_type	body	string	The type of physical network that maps to this network resource. For example, flat, vlan, vxlan, or gre.
segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "segment": {
    "name": null,
    "network_id": "5c0cb560-4089-41dd-be29-469907a23b49",
    "segmentation_id": 2000,
    "network_type": "vlan",
    "physical_network": "segment-1",
    "id": "57fe85e4-ca2f-4192-b3cd-d5c249d7a21f",
    "description": null
  }
}
```

**PUT**

/v2.0/segments/{segment\_id}

## Update segment

Updates a segment.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
segment_id	path	string	The UUID of the segment.
name (Optional)	body	string	Human-readable name of the segment.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "segment": {
    "name": "1",
    "description": "Segment One"
  }
}
```

## Response Parameters

Name	In	Type	Description
id	body	string	The UUID of the segment.
network_id	body	string	The ID of the attached network.
physical_network	body	string	The physical network where this network is implemented.
network_type	body	string	The type of physical network that maps to this network resource. For example, flat, vlan, vxlan,



Name	In	Type	Description
			or gre.
segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "segment": {
    "name": "1",
    "network_id": "5c0cb560-4089-41dd-be29-469907a23b49",
    "segmentation_id": 2000,
    "network_type": "vlan",
    "physical_network": "segment-1",
    "id": "57fe85e4-ca2f-4192-b3cd-d5c249d7a21f",
    "description": "Segment One"
  }
}
```

## DELETE

/v2.0/segments/{segment\_id}

## Delete segment

Deletes a segment and its associated resources.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
segment_id	path	string	The UUID of the segment.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/segments

## List segments

Lists segments to which the project has access.

Use the `fields` query parameter to filter the response. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
id	body	string	The UUID of the segment.
network_id	body	string	The ID of the attached network.
physical_network	body	string	The physical network where this network is implemented.
network_type	body	string	The type of physical network that maps to this network resource. For example, <code>flat</code> , <code>vlan</code> , <code>vxlان</code> , or <code>gre</code> .
segmentation_id	body	integer	The ID of the isolated segment on the physical network. The <code>network_type</code> attribute defines the segmentation model. For example, if the <code>network_type</code> value is <code>vlan</code> , this ID is a <code>vlan</code> identifier. If the <code>network_type</code> value is <code>gre</code> , this ID is a <code>gre</code> key.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "segments": [
    {
      "name": null,
      "network_id": "5c0cb560-4089-41dd-be29-469907a23b49",
      "segmentation_id": 2000,
      "network_type": "vlan",
      "physical_network": "segment-1",
      "id": "57fe85e4-ca2f-4192-b3cd-d5c249d7a21f",
      "description": null
    }
  ]
}
```

```

    "name": null,
    "network_id": "5c0cb560-4089-41dd-be29-469907a23b49",
    "segmentation_id": 2000,
    "network_type": "vlan",
    "physical_network": "segment-2",
    "id": "f1364c3a-4fc1-4206-b2dc-3254bc25cbfc",
    "description": null
  }
}

```

**POST**

/v2.0/segments

## Create segment

Creates a segment.

Normal response codes: 201

Error response codes: 400, 401

### Request

Name	In	Type	Description
network_id	body	string	The ID of the attached network.
physical_network	body	string	The physical network where this network is implemented.
network_type	body	string	The type of physical network that maps to this network resource. For example, flat, vlan, vxlan, or gre.
segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
name (Optional)	body	string	Human-readable name of the segment.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

### Request Example

```

{
  "segment": {
    "network_id": "5c0cb560-4089-41dd-be29-469907a23b49",
    "segmentation_id": 2000,
    "network_type": "vlan",
    "physical_network": "segment-1"
  }
}

```

## Response Parameters

Name	In	Type	Description
id	body	string	The UUID of the segment.
network_id	body	string	The ID of the attached network.
physical_network	body	string	The physical network where this network is implemented.
network_type	body	string	The type of physical network that maps to this network resource. For example, flat, vlan, vxlan, or gre.
segmentation_id	body	integer	The ID of the isolated segment on the physical network. The network_type attribute defines the segmentation model. For example, if the network_type value is vlan, this ID is a vlan identifier. If the network_type value is gre, this ID is a gre key.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.

### Response Example

```
{
  "segment": {
    "name": null,
    "network_id": "5c0cb560-4089-41dd-be29-469907a23b49",
    "segmentation_id": 2000,
    "network_type": "vlan",
    "physical_network": "segment-1",
    "id": "57fe85e4-ca2f-4192-b3cd-d5c249d7a21f",
    "description": null
  }
}
```

## Trunk networking

The trunk extension can be used to multiplex packets coming from and going to multiple neutron logical networks using a single neutron logical port. A trunk is modeled in neutron as a collection of neutron logical ports. One port, called parent port, must be associated to a trunk and it is *the* port to be used to connect instances with neutron. A sequence of subports (or sub\_ports) each typically belonging to distinct neutron networks, is also associated to a trunk, and each subport may have a segmentation type and ID used to mux/demux the traffic coming in and out of the parent port.

In more details, the extension introduces the following resources:

- **trunk.** A top level logical entity to model the group of neutron logical networks whose traffic flows through the trunk.
- **sub\_port.** An association to a neutron logical port with attributes segmentation\_id and segmentation\_type.

GET

/v2.0/trunks

## List trunks

Lists trunks that are accessible to the user who submits the request.

Default policy settings return only those trunks that are owned by the user who submits the request, unless an admin user submits the request.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see the [Filtering](#) section for more details.

Normal response codes: 200

Error response codes: 401, 404

### Response Parameters

Name	In	Type	Description
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
description	body	string	The description for the resource.
id	body	string	The ID for the resource.
name (Optional)	body	string	The name of the resource.
port_id	body	string	The ID of the parent port.
revision_number	body	integer	The revision number of the resource.
status	body	string	The status for the trunk. Possible values are ACTIVE, DOWN, BUILD, DEGRADED, and ERROR.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
sub_ports	body	array	A list of ports associated with the trunk.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).

Name	In	Type	Description
			format).

## Response Example

```
{
  "trunks": [
    {
      "status": "DOWN",
      "sub_ports": [],
      "name": "test",
      "admin_state_up": true,
      "project_id": "313be01bd0744cea86643c711c57012b",
      "tenant_id": "313be01bd0744cea86643c711c57012b",
      "created_at": "2016-10-05T20:11:16Z",
      "updated_at": "2016-10-05T20:11:16Z",
      "revision_number": 1,
      "port_id": "8027c4da-772f-4e43-bfbf-023b4a4e63de",
      "id": "ee98bdb4-a817-43af-943f-4318bff98f51",
      "description": ""
    }
  ]
}
```

**POST**

/v2.0/trunks

## Create trunk

Error codes:

- 400 The operation returns this error code if the request is malformed, e.g. there are missing or invalid parameters in the request.
- 401 The operation is not authorized.
- 404 If the extension is not available or the port UUID of any of the specified ports is not found.
- 409 The operation returns this error code for one of these reasons:
  - A port to be used as subport is in use by another trunk.
  - The segmentation type and segmentation ID are already in use in the trunk.
  - A port to be used as parent port is in use by another trunk or cannot be trunked.
  - A system configuration prevents the operation from succeeding.

Normal response codes: 201

Error response codes: 400, 401, 404, 409

## Request

Name	In	Type	Description
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
port_id	body	string	The ID of the parent port.

Name	In	Type	Description
name (Optional)	body	string	The name of the resource.
description	body	string	The description for the resource.
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
sub_ports	body	array	A list of ports associated with the trunk.

### Request Example

```
{
  "trunk": {
    "port_id": "8027c4da-772f-4e43-bfbf-023b4a4e63de",
    "name": "test",
    "admin_state_up": true
  }
}
```

### Response Parameters

Name	In	Type	Description
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
description	body	string	The description for the resource.
id	body	string	The ID for the resource.
name (Optional)	body	string	The name of the resource.
port_id	body	string	The ID of the parent port.
revision_number	body	integer	The revision number of the resource.
status	body	string	The status for the trunk. Possible values are ACTIVE, DOWN, BUILD, DEGRADED, and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
sub_ports	body	array	A list of ports associated

Name	In	Type	Description
			with the trunk.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).

**PUT**

/v2.0/trunks/{trunk\_id}/add\_subports

## Add subports to trunk

Normal response codes: 200

Error response codes: 400, 401, 404, 409

### Request

Name	In	Type	Description
trunk_id	path	string	The ID of the trunk.
segmentation_id (Optional)	body	integer	The segmentation ID for the subport.
segmentation_type (Optional)	body	string	The segmentation type for the subport.
port_id	body	string	The ID of the subport.

### Request Example

```
{
  "sub_ports": [
    {
      "segmentation_id": 44,
      "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155",
      "segmentation_type": "vlan"
    }
  ]
}
```

### Response Parameters

Name	In	Type	Description
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
description	body	string	The description for the resource.
id	body	string	The ID for the resource.



Name	In	Type	Description
name (Optional)	body	string	The name of the resource.
port_id	body	string	The ID of the parent port.
revision_number	body	integer	The revision number of the resource.
status	body	string	The status for the trunk. Possible values are ACTIVE, DOWN, BUILD, DEGRADED, and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
sub_ports	body	array	A list of ports associated with the trunk.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).

## Response Example

```
{
  "status": "DOWN",
  "sub_ports": [
    {
      "segmentation_type": "vlan",
      "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155",
      "segmentation_id": 44
    }
  ],
  "name": "test",
  "admin_state_up": true,
  "project_id": "145a14e4a64b49bf98baad8945dbd4f1",
  "tenant_id": "145a14e4a64b49bf98baad8945dbd4f1",
  "created_at": "2016-10-05T22:31:37Z",
  "updated_at": "2016-10-05T22:52:04Z",
  "revision_number": 2,
  "port_id": "8027c4da-772f-4e43-bfbf-023b4a4e63de",
  "id": "114a26b1-d124-4835-bb4f-021d3d886023",
  "description": ""
}
```

**PUT**

/v2.0/trunks/{trunk\_id}/remove\_subports

## Delete subports from trunk

Normal response codes: 200

Error response codes: 400, 401, 404, 409

## Request

Name	In	Type	Description
trunk_id	path	string	The ID of the trunk.
port_id	body	string	The ID of the port.

## Request Example

```
{
  "sub_ports": [
    {
      "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155"
    }
  ]
}
```

## Response Parameters

Name	In	Type	Description
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
description	body	string	The description for the resource.
id	body	string	The ID for the resource.
name (Optional)	body	string	The name of the resource.
port_id	body	string	The ID of the parent port.
revision_number	body	integer	The revision number of the resource.
status	body	string	The status for the trunk. Possible values are ACTIVE, DOWN, BUILD, DEGRADED, and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
sub_ports	body	array	A list of ports associated with the trunk.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).

Normal response codes: 200

Error response codes: 401, 404

#### Request

Name	In	Type	Description
trunk_id	path	string	The ID of the trunk.

#### Response Parameters

Name	In	Type	Description
port_id	body	string	The ID of the subport.
segmentation_type (Optional)	body	string	The segmentation type for the subport.
segmentation_id (Optional)	body	integer	The segmentation ID for the subport.

#### Response Example

```
{
  "sub_ports": [
    {
      "segmentation_type": "vlan",
      "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155",
      "segmentation_id": 44
    }
  ]
}
```

**PUT**

/v2.0/trunks/{trunk\_id}

#### Update trunk

The update request is only for changing fields like name, description or admin\_state\_up. Setting the admin\_state\_up to False locks the trunk in that it prevents operations such as adding/removing subports.

Normal response codes: 200

Error response codes: 400, 401, 404, 409

#### Request

Name	In	Type	Description
name_resource (Optional)	body	string	The name of the resource.
admin_state_up_trunk	body	boolean	The administrative state of the resource, which is up (true) or down (false).

Name	In	Type	Description
description_resource	body	string	The description for the resource.
trunk_id	path	string	The ID of the trunk.

## Request Example

```
{
  "trunk": {
    "name": "foo",
    "admin_state_up": true
  }
}
```

## Response Parameters

Name	In	Type	Description
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
description	body	string	The description for the resource.
id	body	string	The ID for the resource.
name (Optional)	body	string	The name of the resource.
port_id	body	string	The ID of the parent port.
revision_number	body	integer	The revision number of the resource.
status	body	string	The status for the trunk. Possible values are ACTIVE, DOWN, BUILD, DEGRADED, and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
sub_ports	body	array	A list of ports associated with the trunk.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).

## Response Example

```
{
  "trunk": {
    "status": "DOWN",
    "sub_ports": [
      {
        "segmentation_type": "vlan",
        "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155",
        "segmentation_id": 44
      }
    ],
    "name": "foo",
    "admin_state_up": true,
    "project_id": "145a14e4a64b49bf98baad8945dbd4f1",
    "tenant_id": "145a14e4a64b49bf98baad8945dbd4f1",
    "created_at": "2016-10-05T22:31:37Z",
    "updated_at": "2016-10-05T23:28:17Z",
    "revision_number": 9,
    "port_id": "8027c4da-772f-4e43-bfbf-023b4a4e63de",
    "id": "114a26b1-d124-4835-bb4f-021d3d886023",
    "description": ""
  }
}
```

**GET**

/v2.0/trunks/{trunk\_id}

## Show trunk

Shows details for a trunk.

Use the `fields` query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
trunk_id	path	string	The ID of the trunk.

## Response Parameters

Name	In	Type	Description
admin_state_up (Optional)	body	boolean	The administrative state of the trunk, which is up (true) or down (false).
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
description	body	string	The description for the resource.
id	body	string	The ID for the resource.

Name	In	Type	Description
name (Optional)	body	string	The name of the resource.
port_id	body	string	The ID of the parent port.
revision_number	body	integer	The revision number of the resource.
status	body	string	The status for the trunk. Possible values are ACTIVE, DOWN, BUILD, DEGRADED, and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
sub_ports	body	array	A list of ports associated with the trunk.
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).

## Response Example

```
{
  "trunk": {
    "status": "DOWN",
    "sub_ports": [
      {
        "segmentation_type": "vlan",
        "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155",
        "segmentation_id": 44
      }
    ],
    "name": "foo",
    "admin_state_up": true,
    "project_id": "145a14e4a64b49bf98baad8945dbd4f1",
    "tenant_id": "145a14e4a64b49bf98baad8945dbd4f1",
    "created_at": "2016-10-05T22:31:37Z",
    "updated_at": "2016-10-05T23:28:17Z",
    "revision_number": 9,
    "port_id": "8027c4da-772f-4e43-bfbf-023b4a4e63de",
    "id": "114a26b1-d124-4835-bb4f-021d3d886023",
    "description": ""
  }
}
```

## DELETE

/v2.0/trunks/{trunk\_id}

## Delete trunk

Deletes a trunk, if its state allows it.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
trunk_id	path	string	The ID of the trunk.

## Trunk details extended attributes (ports)

The `trunk_details` extension attribute is available when showing a port resource that participates in a trunk as parent. The extension is useful for REST clients that may want to access trunk details when getting the parent port, and it allows them to avoid extra lookups.

**GET**

/v2.0/ports/{port\_id}

### Show trunk details

Shows details for a port. The details available in the `trunk_details` attribute contain the trunk ID and the array showing information about the subports that belong to the trunk: the port UUID, the segmentation type, the segmentation ID, and the MAC address.

Normal response codes: 200

## Request

Name	In	Type	Description
port_id	path	string	The ID of the port.

## Response Parameters

Name	In	Type	Description
trunk_details (Optional)	body	dict	The details about the trunk.

## Response Example

```
{
  "port": {
    "status": "DOWN",
    "created_at": "2016-10-05T20:05:14Z",
    "description": "",
    "admin_state_up": true,
    "network_id": "1cf9e069-365f-4a78-8784-616bc12c4c5a",
    "project_id": "313be01bd0744cea86643c711c57012b",
    "tenant_id": "313be01bd0744cea86643c711c57012b",
    "extra_dhcp_opts": [],
    "updated_at": "2016-10-05T20:05:14Z",
    "name": "test",
    "device_owner": "",
    "trunk_details": {
      "trunk_id": "8905d084-010c-46e8-a863-f21cb4441ab1",
```

```

    "sub_ports": [
      {
        "segmentation_id": 33,
        "port_id": "70df9f3e-b409-4761-8304-ce029b2079f5",
        "segmentation_type": "vlan",
        "mac_address": "fa:16:3e:86:9b:dc"
      },
      {
        "segmentation_id": 44,
        "port_id": "4b4c691b-086d-43d2-8a65-5487e9434155",
        "segmentation_type": "vlan",
        "mac_address": "fa:16:3e:fe:29:97"
      }
    ],
    "revision_number": 5,
    "mac_address": "fa:16:3e:5c:e9:a3",
    "fixed_ips": [
      {
        "subnet_id": "76a059c0-b189-479f-882c-5e8bd464ea49",
        "ip_address": "40.0.0.3"
      }
    ],
    "id": "8027c4da-772f-4e43-bfbf-023b4a4e63de",
    "security_groups": ["da88a249-12ac-4221-9565-c406b6feeb48"],
    "device_id": ""
  }
}

```

# Layer 3 Networking

## Floating IPs (floatingips)

**GET**

/v2.0/floatingips

### List floating IPs

Lists floating IPs visible to the user.

Default policy settings return only the floating IPs owned by the user's project, unless the user has admin role.

This example request lists floating IPs in JSON format:

```

GET /v2.0/floatingips
Accept: application/json

```

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401



## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
floatingips	body	array	A list of floatingip objects.
id	body	string	The ID of the floating IP address.
router_id	body	string	The ID of the router for the floating IP.
status	body	string	The status of the floating IP. Values are ACTIVE, DOWN and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).
revision_number	body	integer	The revision number of the resource.
description	body	string	A human-readable description for the resource.
floating_network_id	body	string	The ID of the network associated with the floating IP.
fixed_ip_address	body	string	The fixed IP address that is associated with the floating IP address.
floating_ip_address	body	string	The floating IP address.
port_id	body	string	The ID of a port associated with the floating IP.

## Response Example

```
{
```

```

"floatingips": [
  {
    "router_id": "d23abc8d-2991-4a55-ba98-2aaea84cc72f",
    "description": "for test",
    "created_at": "2016-12-21T10:55:50Z",
    "updated_at": "2016-12-21T10:55:53Z",
    "revision_number": 1,
    "project_id": "4969c491a3c74ee4af974e6d800c62de",
    "tenant_id": "4969c491a3c74ee4af974e6d800c62de",
    "floating_network_id": "376da547-b977-4cfe-9cba-275c80debf57",
    "fixed_ip_address": "10.0.0.3",
    "floating_ip_address": "172.24.4.228",
    "port_id": "ce705c24-c1ef-408a-bda3-7bbd946164ab",
    "id": "2f245a7b-796b-4f26-9cf9-9e82d248fda7",
    "status": "ACTIVE"
  },
  {
    "router_id": null,
    "description": "for test",
    "created_at": "2016-12-21T11:55:50Z",
    "updated_at": "2016-12-21T11:55:53Z",
    "revision_number": 2,
    "project_id": "4969c491a3c74ee4af974e6d800c62de",
    "tenant_id": "4969c491a3c74ee4af974e6d800c62de",
    "floating_network_id": "376da547-b977-4cfe-9cba-275c80debf57",
    "fixed_ip_address": null,
    "floating_ip_address": "172.24.4.227",
    "port_id": null,
    "id": "61cea855-49cb-4846-997d-801b70c71bdd",
    "status": "DOWN"
  }
]
}

```

**POST**

/v2.0/floatingips

## Create floating IP

Creates a floating IP, and, if you specify port information, associates the floating IP with an internal port.

To associate the floating IP with an internal port, specify the port ID attribute in the request body. If you do not specify a port ID in the request, you can issue a PUT request instead of a POST request.

Default policy settings enable only administrative users to set floating IP addresses and some non-administrative users might require a floating IP address. If you do not specify a floating IP address in the request, the operation automatically allocates one.

By default, this operation associates the floating IP address with a single fixed IP address that is configured on an OpenStack Networking port. If a port has multiple IP addresses, you must specify the `fixed_ip_address` attribute in the request body to associate a fixed IP address with the floating IP address.

You can create floating IPs on only external networks. When you create a floating IP, you must specify the ID of the network on which you want to create the floating IP. Alternatively, you can create a floating IP on a subnet in the external network, based on the costs and quality of that subnet.

You must configure an IP address with the internal OpenStack Networking port that

is associated with the floating IP address.

The operation returns the Bad Request (400) response code for one of reasons:

- The network is not external, such as `router:external=False`.
- The internal OpenStack Networking port is not associated with the floating IP address.
- The requested floating IP address does not fall in the subnet range for the external network.
- The fixed IP address is not valid.

If the port ID is not valid, this operation returns 404 response code.

The operation returns the Conflict (409) response code for one of reasons:

- The requested floating IP address is already in use.
- The internal OpenStack Networking port and fixed IP address are already associated with another floating IP.

Normal response codes: 201

Error response codes: 400, 401, 404, 409

## Request

Name	In	Type	Description
floatingip	body	object	A floatingip object. When you associate a floating IP address with a VM, the instance has the same public IP address each time that it boots, basically to maintain a consistent IP address for maintaining DNS assignment.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
floating_network_id	body	string	The ID of the network associated with the floating IP.
fixed_ip_address (Optional)	body	string	The fixed IP address that is associated with the floating IP. If an internal port has multiple associated IP addresses, the service chooses the first IP address unless you explicitly define a fixed IP address in the <code>fixed_ip_address</code> parameter.
floating_ip_address (Optional)	body	string	The floating IP address.
port_id (Optional)	body	string	The ID of a port associated with the floating IP. To associate the floating IP with a fixed IP at creation time, you must specify the identifier of the internal port.
subnet_id (Optional)	body	string	The subnet ID on which you want to create the

Name	In	Type	Description
			floating IP.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

### Request Example

```
{
  "floatingip": {
    "floating_network_id": "376da547-b977-4cfe-9cba-275c80debf57",
    "port_id": "ce705c24-c1ef-408a-bda3-7bbd946164ab",
    "subnet_id": "278d9507-36e7-403c-bb80-1d7093318fe6",
    "fixed_ip_address": "10.0.0.3",
    "floating_ip_address": "172.24.4.228",
    "description": "floating ip for testing"
  }
}
```

### Response Parameters

Name	In	Type	Description
floatingip	body	object	A floatingip object. When you associate a floating IP address with a VM, the instance has the same public IP address each time that it boots, basically to maintain a consistent IP address for maintaining DNS assignment.
router_id	body	string	The ID of the router for the floating IP.
status	body	string	The status of the floating IP. Values are ACTIVE, DOWN and ERROR.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).
revision_number	body	integer	The revision number of the resource.
project_id	body	string	The ID of the project.
floating_network_id	body	string	The ID of the network associated with the floating IP.
fixed_ip_address	body	string	The fixed IP address that is associated with the floating IP address.
floating_ip_address	body	string	The floating IP address.
port_id	body	string	The ID of a port associated with the floating

Name	In	Type	Description
			IP.
id	body	string	The ID of the floating IP address.

**GET**

/v2.0/floatingips/{floatingip\_id}

## Show floating IP details

Shows details for a floating IP.

Use the `fields` query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).

This example request shows details for a floating IP in JSON format. This example also filters the result by the `fixed_ip_address` and `floating_ip_address` fields.

```
GET /v2.0/floatingips/{floatingip_id}?fields=fixed_ip_address
&
fields=floating_ip_address
Accept: application/json
```

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
floatingip_id	path	string	The ID of the floating IP address.

### Response Parameters

Name	In	Type	Description
floatingip	body	object	A floatingip object. When you associate a floating IP address with a VM, the instance has the same public IP address each time that it boots, basically to maintain a consistent IP address for maintaining DNS assignment.
router_id	body	string	The ID of the router for the floating IP.
status	body	string	The status of the floating IP. Values are ACTIVE, DOWN and ERROR.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).

Name	In	Type	Description
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).
revision_number	body	integer	The revision number of the resource.
project_id	body	string	The ID of the project.
floating_network_id	body	string	The ID of the network associated with the floating IP.
fixed_ip_address	body	string	The fixed IP address that is associated with the floating IP address.
floating_ip_address	body	string	The floating IP address.
port_id	body	string	The ID of a port associated with the floating IP.
id	body	string	The ID of the floating IP address.

## Response Example

```
{
  "floatingip": {
    "floating_network_id": "376da547-b977-4cfe-9cba-275c80debf57",
    "router_id": "d23abc8d-2991-4a55-ba98-2aaea84cc72f",
    "fixed_ip_address": "10.0.0.3",
    "floating_ip_address": "172.24.4.228",
    "project_id": "4969c491a3c74ee4af974e6d800c62de",
    "tenant_id": "4969c491a3c74ee4af974e6d800c62de",
    "status": "ACTIVE",
    "port_id": "ce705c24-c1ef-408a-bda3-7bbd946164ab",
    "id": "2f245a7b-796b-4f26-9cf9-9e82d248fda7",
    "description": "floating ip for testing",
    "created_at": "2016-12-21T01:36:04Z",
    "updated_at": "2016-12-21T01:36:04Z",
    "revision_number": 1
  }
}
```

**PUT**

/v2.0/floatingips/{floatingip\_id}

## Update floating IP

Updates a floating IP and its association with an internal port.

The association process is the same as the process for the create floating IP operation.

To disassociate a floating IP from a port, set the port\_id attribute to null or omit it from the request body.

This example updates a floating IP:

PUT /v2.0/floatingips/{floatingip\_id} Accept: application/json

Depending on the request body that you submit, this request associates a port with

or disassociates a port from a floating IP.

Normal response codes: 200

Error response codes: 400, 401, 404, 409

## Request

Name	In	Type	Description
floatingip	body	object	A floatingip object. When you associate a floating IP address with a VM, the instance has the same public IP address each time that it boots, basically to maintain a consistent IP address for maintaining DNS assignment.
floatingip_id	path	string	The ID of the floating IP address.
port_id	body	string	The ID of a port associated with the floating IP. To associate the floating IP with a fixed IP, you must specify the ID of the internal port. To disassociate the floating IP, null should be specified.
fixed_ip_address (Optional)	body	string	The fixed IP address that is associated with the floating IP. If an internal port has multiple associated IP addresses, the service chooses the first IP address unless you explicitly define a fixed IP address in the fixed_ip_address parameter.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "floatingip": {
    "port_id": null
  }
}
```

## Response Parameters

Name	In	Type	Description
floatingip	body	object	A floatingip object. When you associate a floating IP address with a VM, the instance has the same public IP address each time that it boots, basically to maintain a consistent IP address for maintaining DNS

Name	In	Type	Description
			assignment.
router_id	body	string	The ID of the router for the floating IP.
status	body	string	The status of the floating IP. Values are ACTIVE, DOWN and ERROR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
floating_network_id	body	string	The ID of the network associated with the floating IP.
fixed_ip_address	body	string	The fixed IP address that is associated with the floating IP address.
floating_ip_address	body	string	The floating IP address.
port_id	body	string	The ID of a port associated with the floating IP.
id	body	string	The ID of the floating IP address.
created_at	body	string	Time at which the resource has been created (in UTC ISO8601 format).
updated_at	body	string	Time at which the resource has been updated (in UTC ISO8601 format).
revision_number	body	integer	The revision number of the resource.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "floatingip": {
    "floating_network_id": "376da547-b977-4cfe-9cba-275c80debf57",
    "router_id": "d23abc8d-2991-4a55-ba98-2aaea84cc72f",
    "fixed_ip_address": null,
    "floating_ip_address": "172.24.4.228",
    "project_id": "4969c491a3c74ee4af974e6d800c62de",
    "tenant_id": "4969c491a3c74ee4af974e6d800c62de",
    "status": "ACTIVE",
    "port_id": null,
    "id": "2f245a7b-796b-4f26-9cf9-9e82d248fda7",
    "description": "for test",
    "created_at": "2016-12-21T10:55:50Z",
    "updated_at": "2016-12-22T03:13:49Z",
    "revision_number": 3
  }
}
```



```
}  
}
```

## DELETE

/v2.0/floatingips/{floatingip\_id}

### Delete floating IP

Deletes a floating IP and, if present, its associated port.

This example deletes a floating IP:

```
DELETE /v2.0/floatingips/{floatingip_id} Accept: application/json
```

Normal response codes: 204

Error response codes: 401, 404

#### Request

Name	In	Type	Description
floatingip_id	path	string	The ID of the floating IP address.

#### Response

There is no body content for the response of a successful DELETE request.

## Routers (routers)

A router is a logical entity for forwarding packets across internal subnets and NATting them on external networks through an appropriate external gateway.

This resource is provided when router extension is enabled.

## GET

/v2.0/routers

### List routers

Lists logical routers that the project who submits the request can access.

Default policy settings return only those routers that the project who submits the request owns, unless an administrative user submits the request.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
routers	body	array	A list of router objects.
id	body	string	The ID of the router.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
status	body	string	The router status.
external_gateway_info	body	object	The external gateway information of the router. If the router has an external gateway, this would be a dict with <code>network_id</code> , <code>enable_snat</code> and <code>external_fixed_ips</code> . Otherwise, this would be null.
network_id	body	string	Network ID which the router gateway is connected to.

Name	In	Type	Description
enable_snat	body	boolean	Enable Source NAT (SNAT) attribute. true means Network Address Translation (NAT) is enabled for traffic generated by subnets attached to the router when the traffic is sent to/received from the external network. false means no NAT is applied for traffic from/to the external network. It is available when ext-gw-mode extension is enabled.
external_fixed_ips	body	array	IP address(es) of the external gateway of the router. It is a list of IP addresses. Each element of the list is a dictionary of ip_address and subnet_id.
routes	body	array	The extra routes configuration for L3 router. A list of dictionaries with destination and nexthop parameters. It is available when extraroute extension is enabled.
destination	body	string	The destination CIDR.
nexthop	body	string	The IP address of the next hop for the corresponding destination. The next hop IP address must be a part of one of the subnets to which the router interfaces are connected.
distributed	body	boolean	true indicates a distributed router. It is available when dvr extension is enabled.
ha	body	boolean	true indicates a highly-available router. It is available when l3-ha extension is enabled.
availability_zone_hints	body	array	The availability zone candidates for the router. It is available when router_availability_zone extension is enabled.
availability_zones	body	array	The availability zone(s) for the router. It is available when router_availability_zone extension is enabled.

## Response Example

```
{
```

```

"routers": [
  {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "description": "",
    "distributed": false,
    "external_gateway_info": {
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.3",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        },
        {
          "ip_address": "2001:db8::c",
          "subnet_id": "0c56df5d-ace5-46c8-8f4c-45fa4e334d18"
        }
      ]
    },
    "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3"
  },
  {
    "ha": false,
    "id": "915a14a6-867b-4af7-83d1-70efceb146f9",
    "name": "router2",
    "routes": [],
    "status": "ACTIVE",
    "project_id": "0bd18306d801447bb457a46252d82d13",
    "tenant_id": "0bd18306d801447bb457a46252d82d13"
  },
  {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "description": "",
    "distributed": false,
    "external_gateway_info": {
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.6",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        },
        {
          "ip_address": "2001:db8::9",
          "subnet_id": "0c56df5d-ace5-46c8-8f4c-45fa4e334d18"
        }
      ]
    },
    "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3"
  },
  {
    "ha": false,
    "id": "f8a44de0-fc8e-45df-93c7-f79bf3b01c95",
    "name": "router1",
    "routes": [],
    "status": "ACTIVE",
    "project_id": "0bd18306d801447bb457a46252d82d13",
    "tenant_id": "0bd18306d801447bb457a46252d82d13"
  }
]
}

```

**POST**

/v2.0/routers

## Create router

Creates a logical router.

This operation creates a logical router. The logical router does not have any internal interface and it is not associated with any subnet. You can optionally specify an external gateway for a router at create time. The external gateway for the router must be plugged into an external network. An external network has its `router:external` extended field set to `true`. To specify an external gateway, the ID of the external network must be passed in the `network_id` parameter of the `external_gateway_info` attribute in the request body.

Normal response codes: 201

Error response codes: 400, 401

### Request

Name	In	Type	Description
router	body	object	A router object.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up ( <code>true</code> ) or down ( <code>false</code> ). Default is <code>true</code> .
external_gateway_info (Optional)	body	object	The external gateway information of the router. If the router has an external gateway, this would be a dict with <code>network_id</code> , <code>enable_snat</code> and <code>external_fixed_ips</code> . Otherwise, this would be <code>null</code> .
network_id	body	string	Network ID which the router gateway is connected to.
enable_snat (Optional)	body	boolean	Enable Source NAT (SNAT) attribute. Default is <code>true</code> . To persist this attribute value, set the <code>enable_snat_by_default</code> option in the <code>neutron.conf</code> file. It is available when ext-

Name	In	Type	Description
			gw-mode extension is enabled.
external_fixed_ips (Optional)	body	array	IP address(es) of the external gateway interface of the router. It is a list of IP addresses you would like to assign to the external gateway interface. Each element of this list is a dictionary of ip_address and subnet_id.
distributed (Optional)	body	boolean	true indicates a distributed router. It is available when dvr extension is enabled.
ha (Optional)	body	boolean	true indicates a highly-available router. It is available when l3-ha extension is enabled.
availability_zone_hints (Optional)	body	array	The availability zone candidates for the router. It is available when router_availability_zone extension is enabled.

### Request Example

```
{
  "router": {
    "name": "router1",
    "external_gateway_info": {
      "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3",
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.6",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        }
      ]
    },
    "admin_state_up": true
  }
}
```

### Response Parameters

Name	In	Type	Description
router	body	object	A router object.
id	body	string	The ID of the router.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
admin_state_up	body	boolean	The administrative state of the resource,

Name	In	Type	Description
			which is up (true) or down (false).
status	body	string	The router status.
external_gateway_info	body	object	The external gateway information of the router. If the router has an external gateway, this would be a dict with network_id, enable_snat and external_fixed_ips. Otherwise, this would be null.
network_id	body	string	Network ID which the router gateway is connected to.
enable_snat	body	boolean	Enable Source NAT (SNAT) attribute. true means Network Address Translation (NAT) is enabled for traffic generated by subnets attached to the router when the traffic is sent to/received from the external network. false means no NAT is applied for traffic from/to the external network. It is available when ext-gw-mode extension is enabled.
external_fixed_ips	body	array	IP address(es) of the external gateway of the router. It is a list of IP addresses. Each element of the list is a dictionary of ip_address and subnet_id.
routes	body	array	The extra routes configuration for L3 router. A list of dictionaries with destination and nexthop parameters. It is available when extraroute extension is enabled.
destination	body	string	The destination CIDR.
nexthop	body	string	The IP address of the next hop for the corresponding destination. The next hop IP address must be a part of one of the subnets to which the router interfaces are connected.
distributed	body	boolean	true indicates a distributed router. It is available when dvr extension is enabled.
ha	body	boolean	true indicates a highly-available router. It is available when l3-ha extension is enabled.
availability_zone_hints	body	array	The availability zone candidates for the router. It is available when router_availability_zone extension is enabled.

Name	In	Type	Description
availability_zones	body	array	The availability zone(s) for the router. It is available when router_availability_zone extension is enabled.

## Response Example

```
{
  "router": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "description": "",
    "distributed": false,
    "external_gateway_info": {
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.6",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        }
      ]
    },
    "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3"
  },
  "ha": false,
  "id": "f8a44de0-fc8e-45df-93c7-f79bf3b01c95",
  "name": "router1",
  "routes": [],
  "status": "ACTIVE",
  "project_id": "0bd18306d801447bb457a46252d82d13",
  "tenant_id": "0bd18306d801447bb457a46252d82d13"
}
```

**GET**

/v2.0/routers/{router\_id}

## Show router details

Shows details for a router.

Use the fields query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
router_id	path	string	The ID of the router.
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings.



Name	In	Type	Description
			By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
router	body	object	A router object.
id	body	string	The ID of the router.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up ( <code>true</code> ) or down ( <code>false</code> ).
status	body	string	The router status.
external_gateway_info	body	object	The external gateway information of the router. If the router has an external gateway, this would be a dict with <code>network_id</code> , <code>enable_snat</code> and <code>external_fixed_ips</code> . Otherwise, this would be null.
network_id	body	string	Network ID which the router gateway is connected to.
enable_snat	body	boolean	Enable Source NAT (SNAT) attribute. <code>true</code> means Network Address Translation (NAT) is enabled for traffic generated by subnets attached to the router when the traffic is sent to/received from the external network. <code>false</code> means no NAT is applied for traffic from/to the external network. It is available when <code>ext-gw-mode</code> extension is enabled.
external_fixed_ips	body	array	IP address(es) of the external gateway of the router. It is a list of IP addresses. Each element of the list is a dictionary of <code>ip_address</code> and <code>subnet_id</code> .
routes	body	array	The extra routes configuration for L3 router. A list of dictionaries with

Name	In	Type	Description
			destination and nexthop parameters. It is available when extraroute extension is enabled.
destination	body	string	The destination CIDR.
nexthop	body	string	The IP address of the next hop for the corresponding destination. The next hop IP address must be a part of one of the subnets to which the router interfaces are connected.
distributed	body	boolean	true indicates a distributed router. It is available when dvr extension is enabled.
ha	body	boolean	true indicates a highly-available router. It is available when l3-ha extension is enabled.
availability_zone_hints	body	array	The availability zone candidates for the router. It is available when router_availability_zone extension is enabled.
availability_zones	body	array	The availability zone(s) for the router. It is available when router_availability_zone extension is enabled.

## Response Example

```
{
  "router": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "description": "",
    "distributed": false,
    "external_gateway_info": {
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.6",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        },
        {
          "ip_address": "2001:db8::9",
          "subnet_id": "0c56df5d-ace5-46c8-8f4c-45fa4e334d18"
        }
      ]
    },
    "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3"
  },
  "ha": false,
  "id": "f8a44de0-fc8e-45df-93c7-f79bf3b01c95",
  "name": "router1",
  "routes": [],
  "status": "ACTIVE",
  "project_id": "0bd18306d801447bb457a46252d82d13",
  "tenant_id": "0bd18306d801447bb457a46252d82d13"
}
```

```
}  
}
```

**PUT**

/v2.0/routers/{router\_id}

## Update router

Updates a logical router.

This operation does not enable the update of router interfaces. To update a router interface, use the add router interface and remove router interface operations.

Normal response codes: 200

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
router	body	object	A router object.
external_gateway_info	body	object	The external gateway information of the router. If the router has an external gateway, this would be a dict with network_id, enable_snat and external_fixed_ips. Otherwise, this would be null.
enable_snat	body	boolean	Enable Source NAT (SNAT) attribute. true means Network Address Translation (NAT) is enabled for traffic generated by subnets attached to the router when the traffic is sent to/received from the external network. false means no NAT is applied for traffic from/to the external network. It is available when ext-gw-mode extension is enabled.
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
external_fixed_ips	body	array	IP address(es) of the external gateway of the router. It is a list of IP addresses. Each element of the list is a dictionary of ip_address and subnet_id.
router_id	path	string	The ID of the router.

Name	In	Type	Description
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "router": {
    "external_gateway_info": {
      "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3",
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.6",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        }
      ]
    }
  }
}
```

## Response Parameters

Name	In	Type	Description
router	body	object	A router object.
id	body	string	The ID of the router.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
status	body	string	The router status.
external_gateway_info	body	object	The external gateway information of the router. If the router has an external gateway, this would be a dict with network_id, enable_snat and external_fixed_ips. Otherwise, this would be null.
network_id	body	string	Network ID which the router gateway is connected to.
enable_snat	body	boolean	Enable Source NAT (SNAT) attribute. true means Network Address Translation (NAT) is enabled for traffic generated by subnets attached to the router when the traffic is sent to/received from the external network.

Name	In	Type	Description
			false means no NAT is applied for traffic from/to the external network. It is available when ext-gw-mode extension is enabled.
external_fixed_ips	body	array	IP address(es) of the external gateway of the router. It is a list of IP addresses. Each element of the list is a dictionary of ip_address and subnet_id.
routes	body	array	The extra routes configuration for L3 router. A list of dictionaries with destination and nexthop parameters. It is available when extraroute extension is enabled.
destination	body	string	The destination CIDR.
nexthop	body	string	The IP address of the next hop for the corresponding destination. The next hop IP address must be a part of one of the subnets to which the router interfaces are connected.
distributed	body	boolean	true indicates a distributed router. It is available when dvr extension is enabled.
ha	body	boolean	true indicates a highly-available router. It is available when l3-ha extension is enabled.
availability_zone_hints	body	array	The availability zone candidates for the router. It is available when router_availability_zone extension is enabled.
availability_zones	body	array	The availability zone(s) for the router. It is available when router_availability_zone extension is enabled.

## Response Example

```
{
  "router": {
    "admin_state_up": true,
    "availability_zone_hints": [],
    "availability_zones": [
      "nova"
    ],
    "description": "",
    "distributed": false,
    "external_gateway_info": {
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.24.4.6",
          "subnet_id": "b930d7f6-ceb7-40a0-8b81-a425dd994ccf"
        }
      ]
    },
    "network_id": "ae34051f-aa6c-4c75-abf5-50dc9ac99ef3"
  },
  "ha": false,
  "id": "f8a44de0-fc8e-45df-93c7-f79bf3b01c95",
}
```

```
    "name": "router1",
    "routes": [],
    "status": "ACTIVE",
    "project_id": "0bd18306d801447bb457a46252d82d13",
    "tenant_id": "0bd18306d801447bb457a46252d82d13"
  }
}
```

## DELETE

/v2.0/routers/{router\_id}

### Delete router

Deletes a logical router and, if present, its external gateway interface.

This operation fails if the router has attached interfaces. Use the remove router interface operation to remove all router interfaces before you delete the router.

Normal response codes: 204

Error response codes: 401, 404, 409

#### Request

Name	In	Type	Description
router_id	path	string	The ID of the router.

#### Response

There is no body content for the response of a successful DELETE request.

## PUT

/v2.0/routers/{router\_id}/add\_router\_interface

### Add interface to router

Adds an internal interface to a logical router. This means a specified subnet is attached to a router as an internal router interface.

Specify the ID of a subnet or port in the request body:

- Subnet ID. The gateway IP address for the subnet is used as an IP address of the created router interface.
- Port ID. The IP address associated with the port is used as an IP address of the created router interface.

When you specify an IPv6 subnet, this operation adds the subnet to an existing internal port with same network ID, on the router. If a port with the same network ID does not exist, this operation creates a port on the router for that subnet.

The limitation of one IPv4 subnet per router port remains, though a port can contain any number of IPv6 subnets that belong to the same network ID.

When you use the port-create command to add a port and then call router-interface-add with this port ID, this operation adds the port to the router if the following conditions are met:

- The port has no more than one IPv4 subnet.
- The IPv6 subnets, if any, on the port do not have same network ID as the network ID of IPv6 subnets on any other ports.

If you specify both subnet ID and port ID, this operation returns the Bad Request (400) response code.

If the port is already in use, this operation returns the Conflict (409) response code.

This operation returns a port ID that is either:

- The same ID that is passed in the request body when a port is specified.
- The ID of a port that this operation creates to attach the subnet to the router.

After you run this operation, the operation sets:

- The `device_id` attribute of this port to the router ID
- The `device_owner` attribute to `network:router_interface`

Normal response codes: 200

Error response codes: 400, 401, 404, 409

## Request

Name	In	Type	Description
router_id	path	string	The ID of the router.
subnet_id (Optional)	body	string	The ID of the subnet. One of subnet_id or port_id must be specified.
port_id (Optional)	body	string	The ID of the port. One of subnet_id or port_id must be specified.

## Request Example

```
{
  "subnet_id": "a2f1f29d-571b-4533-907f-5803ab96ead1"
}
```

or

```
{
  "port_id": "2dc46bcc-d1f2-4077-b99e-91ee28afaff0"
}
```

## Response Parameters

Name	In	Type	Description
id	body	string	The ID of the router.
subnet_id	body	string	The ID of the subnet which the

Name	In	Type	Description
			router interface belongs to.
subnet_ids	body	array	A list of the ID of the subnet which the router interface belongs to. The list contains only one member.
tenant_id	body	string	The ID of the project who owns the router interface.
project_id	body	string	The ID of the project who owns the router interface.
port_id	body	string	The ID of the port which represents the router interface.
network_id	body	string	Network ID which the router interface is connected to.

## Response Example

```
{
  "id": "915a14a6-867b-4af7-83d1-70efceb146f9",
  "network_id": "91c013e2-d65a-474e-9177-c3e1799ca726",
  "port_id": "2dc46bcc-d1f2-4077-b99e-91ee28afaff0",
  "subnet_id": "a2f1f29d-571b-4533-907f-5803ab96ead1",
  "subnet_ids": [
    "a2f1f29d-571b-4533-907f-5803ab96ead1"
  ],
  "project_id": "0bd18306d801447bb457a46252d82d13",
  "tenant_id": "0bd18306d801447bb457a46252d82d13"
}
```

**PUT**

/v2.0/routers/{router\_id}/remove\_router\_interface

## Remove interface from router

Deletes an internal interface from a logical router.

This operation deletes an internal router interface, which detaches a subnet from the router. If this subnet ID is the last subnet on the port, this operation deletes the port itself. You must specify either a subnet ID or port ID in the request body; the operation uses this value to identify which router interface to deletes.

You can also specify both a subnet ID and port ID. If you specify both IDs, the subnet ID must correspond to the subnet ID of the first IP address on the port. Otherwise, this operation returns the **Conflict** (409) response code with information about the affected router and interface.

If you try to delete the router interface for subnets that are used by one or more routes, this operation returns the **Conflict** (409) response. In this case, you first need to delete such routes from the router.

If the router or the subnet and port do not exist or are not visible to you, this operation returns the **Not Found** (404) response code. As a consequence of this



operation, the operation removes the port connecting the router with the subnet from the subnet for the network.

Normal response codes: 200

Error response codes: 400, 401, 404, 409

### Request

Name	In	Type	Description
router_id	path	string	The ID of the router.
subnet_id (Optional)	body	string	The ID of the subnet. One of subnet_id or port_id must be specified.
port_id (Optional)	body	string	The ID of the port. One of subnet_id or port_id must be specified.

### Request Example

```
{  
  "subnet_id": "a2f1f29d-571b-4533-907f-5803ab96ead1"  
}
```

or

```
{  
  "port_id": "2dc46bcc-d1f2-4077-b99e-91ee28afaff0"  
}
```

### Response Parameters

Name	In	Type	Description
id	body	string	The ID of the router.
subnet_id	body	string	The ID of the subnet which the router interface belongs to.
subnet_ids	body	array	A list of the ID of the subnet which the router interface belongs to. The list contains only one member.
tenant_id	body	string	The ID of the project who owns the router interface.
project_id	body	string	The ID of the project who owns the router interface.
port_id	body	string	The ID of the port which represents the router interface.
network_id	body	string	Network ID which the router interface is connected to.

## Response Parameters

Name	In	Type	Description
subnetpool	body	object	A subnetpool object.
id	body	string	The ID of the subnet.
name	body	string	Human-readable name of the resource.
default_quota (Optional)	body	integer	A per-project quota on the prefix space that can be allocated from the subnet pool for project subnets. Default is no quota is enforced on allocations from the subnet pool. For IPv4 subnet pools, default_quota is measured in units of /32. For IPv6 subnet pools, default_quota is measured units of /64. All projects that use the subnet pool have the same prefix quota applied.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
created_at	body	string	Time at which port has been created.
updated_at	body	string	Time at which port has been updated.
prefixes	body	array	A list of subnet prefixes to assign to the subnet pool. The API merges adjacent prefixes and treats them as a single prefix. Each subnet prefix must be unique among all subnet prefixes in all subnet pools that are associated with the address scope.
min_prefixlen (Optional)	body	integer	The smallest prefix that can be allocated from a subnet pool. For IPv4 subnet pools, default is 8. For IPv6 subnet pools, default is 64.
address_scope_id (Optional)	body	string	An address scope to assign to the subnet pool.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
default_prefixlen (Optional)	body	integer	The size of the prefix to allocate when the cidr or prefixlen attributes are omitted when you create the subnet. Default is min_prefixlen.
max_prefixlen (Optional)	body	integer	The maximum prefix size that can be allocated from the subnet pool. For IPv4 subnet pools, default is 32. For IPv6 subnet pools, default is 128.

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
is_default	body	boolean	The subnetpool is default pool or not.
revision_number	body	string	The revision number of the subnetpool.

## Response Example

```
{
  "subnetpool": {
    "min_prefixlen": "64",
    "address_scope_id": null,
    "default_prefixlen": "64",
    "id": "03f761e6-eee0-43fc-a921-8acf64c14988",
    "max_prefixlen": "64",
    "name": "my-subnet-pool",
    "default_quota": null,
    "is_default": false,
    "project_id": "9fadcee8aa7c40cdb2114fff7d569c08",
    "tenant_id": "9fadcee8aa7c40cdb2114fff7d569c08",
    "created_at": "2016-03-08T20:19:41",
    "prefixes": [
      "2001:db8:0:2::/64",
      "2001:db8::/63"
    ],
    "updated_at": "2016-03-08T20:19:41",
    "ip_version": 6,
    "shared": false,
    "description": "",
    "revision_number": 2
  }
}
```

**PUT**

/v2.0/subnetpools/{subnetpool\_id}

## Update subnet pool

Updates a subnet pool.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
subnetpool_id	path	string	The UUID of the subnet pool.
subnetpool	body	object	A subnetpool object.
name	body	string	Human-readable name of the resource.
default_quota (Optional)	body	integer	A per-project quota on the prefix space that can be allocated from the

Name	In	Type	Description
			subnet pool for project subnets. Default is no quota is enforced on allocations from the subnet pool. For IPv4 subnet pools, default_quota is measured in units of /32. For IPv6 subnet pools, default_quota is measured units of /64. All projects that use the subnet pool have the same prefix quota applied.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
prefixes	body	array	A list of subnet prefixes to assign to the subnet pool. The API merges adjacent prefixes and treats them as a single prefix. Each subnet prefix must be unique among all subnet prefixes in all subnet pools that are associated with the address scope.
min_prefixlen (Optional)	body	integer	The smallest prefix that can be allocated from a subnet pool. For IPv4 subnet pools, default is 8. For IPv6 subnet pools, default is 64.
address_scope_id (Optional)	body	string	An address scope to assign to the subnet pool.
default_prefixlen (Optional)	body	integer	The size of the prefix to allocate when the cidr or prefixlen attributes are omitted when you create the subnet. Default is min_prefixlen.
max_prefixlen (Optional)	body	integer	The maximum prefix size that can be allocated from the subnet pool. For IPv4 subnet pools, default is 32. For IPv6 subnet pools, default is 128.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

### Request Example

```
{
  "subnetpool": {
    "name": "my-new-subnetpool-name",
    "prefixes": [
      "2001:db8::/64",
      "2001:db8:0:1::/64",
      "2001:db8:0:2::/64"
    ],
    "min_prefixlen": 64,
    "default_prefixlen": 64,
    "max_prefixlen": 64
  }
}
```

## Response Parameters

Name	In	Type	Description
subnetpool	body	object	A subnetpool object.
id	body	string	The ID of the subnet.
name	body	string	Human-readable name of the resource.
default_quota (Optional)	body	integer	A per-project quota on the prefix space that can be allocated from the subnet pool for project subnets. Default is no quota is enforced on allocations from the subnet pool. For IPv4 subnet pools, default_quota is measured in units of /32. For IPv6 subnet pools, default_quota is measured units of /64. All projects that use the subnet pool have the same prefix quota applied.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
created_at	body	string	Time at which port has been created.
updated_at	body	string	Time at which port has been updated.
prefixes	body	array	A list of subnet prefixes to assign to the subnet pool. The API merges adjacent prefixes and treats them as a single prefix. Each subnet prefix must be unique among all subnet prefixes in all subnet pools that are associated with the address scope.
min_prefixlen (Optional)	body	integer	The smallest prefix that can be allocated from a subnet pool. For IPv4 subnet pools, default is 8. For IPv6 subnet pools, default is 64.
address_scope_id (Optional)	body	string	An address scope to assign to the subnet pool.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.

Name	In	Type	Description
default_prefixlen (Optional)	body	integer	The size of the prefix to allocate when the cidr or prefixlen attributes are omitted when you create the subnet. Default is min_prefixlen.
max_prefixlen (Optional)	body	integer	The maximum prefix size that can be allocated from the subnet pool. For IPv4 subnet pools, default is 32. For IPv6 subnet pools, default is 128.
description	body	string	A human-readable description for the resource.
is_default	body	boolean	The subnetpool is default pool or not.
revision_number	body	string	The revision number of the subnetpool.

## Response Example

```
{
  "subnetpool": {
    "name": "my-new-subnetpool-name",
    "default_quota": null,
    "is_default": false,
    "project_id": "9fadcee8aa7c40cdb2114fff7d569c08",
    "tenant_id": "9fadcee8aa7c40cdb2114fff7d569c08",
    "prefixes": [
      "2001:db8::/63",
      "2001:db8:0:2::/64"
    ],
    "min_prefixlen": 64,
    "address_scope_id": null,
    "ip_version": 6,
    "shared": false,
    "default_prefixlen": 64,
    "id": "03f761e6-eee0-43fc-a921-8acf64c14988",
    "max_prefixlen": 64,
    "description": "",
    "revision_number": 2
  }
}
```

## DELETE

/v2.0/subnetpools/{subnetpool\_id}

## Delete subnet pool

Deletes a subnet pool.

The operation fails if any subnets allocated from the subnet pool are still in use.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
subnetpool_id	path	string	The UUID of the subnet pool.

## Response

There is no body content for the response of a successful DELETE request.

### GET

/v2.0/subnetpools

## List subnet pools

Lists subnet pools that the project has access to.

Default policy settings return only the subnet pools owned by the project of the user submitting the request, unless the user has administrative role.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
subnetpools	body	array	A list of subnetpool objects.
id	body	string	The ID of the subnet.

Name	In	Type	Description
name	body	string	Human-readable name of the resource.
default_quota (Optional)	body	integer	A per-project quota on the prefix space that can be allocated from the subnet pool for project subnets. Default is no quota is enforced on allocations from the subnet pool. For IPv4 subnet pools, default_quota is measured in units of /32. For IPv6 subnet pools, default_quota is measured units of /64. All projects that use the subnet pool have the same prefix quota applied.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
created_at	body	string	Time at which port has been created.
updated_at	body	string	Time at which port has been updated.
prefixes	body	array	A list of subnet prefixes to assign to the subnet pool. The API merges adjacent prefixes and treats them as a single prefix. Each subnet prefix must be unique among all subnet prefixes in all subnet pools that are associated with the address scope.
min_prefixlen (Optional)	body	integer	The smallest prefix that can be allocated from a subnet pool. For IPv4 subnet pools, default is 8. For IPv6 subnet pools, default is 64.
address_scope_id (Optional)	body	string	An address scope to assign to the subnet pool.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
default_prefixlen (Optional)	body	integer	The size of the prefix to allocate when the cidr or prefixlen attributes are omitted when you create the subnet. Default is



Name	In	Type	Description
			min_prefixlen.
max_prefixlen (Optional)	body	integer	The maximum prefix size that can be allocated from the subnet pool. For IPv4 subnet pools, default is 32. For IPv6 subnet pools, default is 128.
description	body	string	A human-readable description for the resource.
is_default	body	boolean	The subnetpool is default pool or not.
revision_number	body	string	The revision number of the subnetpool.

## Response Example

```
{
  "subnetpools": [
    {
      "min_prefixlen": "64",
      "address_scope_id": null,
      "default_prefixlen": "64",
      "id": "03f761e6-eee0-43fc-a921-8acf64c14988",
      "max_prefixlen": "64",
      "name": "my-subnet-pool-ipv6",
      "default_quota": null,
      "is_default": false,
      "project_id": "9fadcee8aa7c40cdb2114fff7d569c08",
      "tenant_id": "9fadcee8aa7c40cdb2114fff7d569c08",
      "prefixes": [
        "2001:db8:0:2::/64",
        "2001:db8::/63"
      ],
      "ip_version": 6,
      "shared": false,
      "description": "",
      "revision_number": 2
    },
    {
      "min_prefixlen": "24",
      "address_scope_id": null,
      "default_prefixlen": "25",
      "id": "f49a1319-423a-4ee6-ba54-1d95a4f6cc68",
      "max_prefixlen": "30",
      "name": "my-subnet-pool-ipv4",
      "default_quota": null,
      "is_default": false,
      "project_id": "9fadcee8aa7c40cdb2114fff7d569c08",
      "tenant_id": "9fadcee8aa7c40cdb2114fff7d569c08",
      "prefixes": [
        "10.10.0.0/21",
        "192.168.0.0/16"
      ],
      "ip_version": 4,
      "shared": false,
      "description": "",
      "revision_number": 2
    }
  ]
}
```

**POST**

/v2.0/subnetpools

## Create subnet pool

Creates a subnet pool.

Normal response codes: 201

Error response codes: 400, 401, 403, 404

### Request

Name	In	Type	Description
subnetpool	body	object	A subnetpool object.
name	body	string	Human-readable name of the resource.
default_quota (Optional)	body	integer	A per-project quota on the prefix space that can be allocated from the subnet pool for project subnets. Default is no quota is enforced on allocations from the subnet pool. For IPv4 subnet pools, default_quota is measured in units of /32. For IPv6 subnet pools, default_quota is measured units of /64. All projects that use the subnet pool have the same prefix quota applied.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
prefixes	body	array	A list of subnet prefixes to assign to the subnet pool. The API merges adjacent prefixes and treats them as a single prefix. Each subnet prefix must be unique among all subnet prefixes in all subnet pools that are associated with the address scope.
min_prefixlen (Optional)	body	integer	The smallest prefix that can be allocated from a subnet pool. For IPv4 subnet pools, default is 8. For IPv6 subnet pools, default is 64.
address_scope_id (Optional)	body	string	An address scope to assign to the subnet pool.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
default_prefixlen	body	integer	The size of the prefix to allocate

Name	In	Type	Description
(Optional)			when the cidr or prefixlen attributes are omitted when you create the subnet. Default is min_prefixlen.
max_prefixlen (Optional)	body	integer	The maximum prefix size that can be allocated from the subnet pool. For IPv4 subnet pools, default is 32. For IPv6 subnet pools, default is 128.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

### Request Example

```
{
  "subnetpool": {
    "name": "my-subnet-pool",
    "prefixes": [
      "192.168.0.0/16",
      "10.10.0.0/21"
    ],
    "default_prefixlen": 25,
    "min_prefixlen": 24,
    "max_prefixlen": 30,
    "shared": "false"
  }
}
```

### Response Parameters

Name	In	Type	Description
subnetpool	body	object	A subnetpool object.
id	body	string	The ID of the subnet.
name	body	string	Human-readable name of the resource.
default_quota (Optional)	body	integer	A per-project quota on the prefix space that can be allocated from the subnet pool for project subnets. Default is no quota is enforced on allocations from the subnet pool. For IPv4 subnet pools, default_quota is measured in units of /32. For IPv6 subnet pools, default_quota is measured units of /64. All projects that use the subnet pool have the same prefix quota applied.
tenant_id	body	string	The ID of the project.

Name	In	Type	Description
project_id	body	string	The ID of the project.
created_at	body	string	Time at which port has been created.
updated_at	body	string	Time at which port has been updated.
prefixes	body	array	A list of subnet prefixes to assign to the subnet pool. The API merges adjacent prefixes and treats them as a single prefix. Each subnet prefix must be unique among all subnet prefixes in all subnet pools that are associated with the address scope.
min_prefixlen (Optional)	body	integer	The smallest prefix that can be allocated from a subnet pool. For IPv4 subnet pools, default is 8. For IPv6 subnet pools, default is 64.
address_scope_id (Optional)	body	string	An address scope to assign to the subnet pool.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
default_prefixlen (Optional)	body	integer	The size of the prefix to allocate when the cidr or prefixlen attributes are omitted when you create the subnet. Default is min_prefixlen.
max_prefixlen (Optional)	body	integer	The maximum prefix size that can be allocated from the subnet pool. For IPv4 subnet pools, default is 32. For IPv6 subnet pools, default is 128.
description	body	string	A human-readable description for the resource.
is_default	body	boolean	The subnetpool is default pool or not.
revision_number	body	string	The revision number of the subnetpool.

# Subnets

Lists, shows details for, creates, updates, and deletes subnet resources.

**GET**

/v2.0/subnets

## List subnets

Lists subnets that the project has access to.

Default policy settings return only subnets owned by the project of the user submitting the request, unless the user has administrative role. You can control which attributes are returned by using the fields query parameter. You can filter results by using query string parameters.

Use the fields query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using fields parameter, the API returns only the requested set of attributes. fields parameter can be specified multiple times. For example, if you specify fields=id&fields=name in the request URL, only id and name attributes will be returned.

## Response Parameters

Name	In	Type	Description
subnets	body	array	A list of subnet objects.
id	body	string	The ID of the subnet.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.

Name	In	Type	Description
name	body	string	Human-readable name of the resource.
enable_dhcp	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers	body	array	List of dns name servers associated with the subnet.
allocation_pools	body	array	Allocation pools with start and end IP addresses for this subnet.
host_routes	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet.
cidr	body	string	The CIDR of the subnet.
created_at	body	string	Time at which the subnet has been created.
description	body	string	A human-readable description for the resource.
ipv6_address_mode	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
ipv6_ra_mode	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
revision_number	body	string	The revision number of the subnet.
segment_id	body	string	The ID of a network segment the subnet is associated with.

Name	In	Type	Description
			It is available when segment extension is enabled.
service_types	body	string	The service types associated with the subnet.
subnetpool_id	body	string	The ID of the subnet pool associated with the subnet.
updated_at	body	string	Time at which the subnet has been updated.

## Response Example

```
{
  "subnets": [
    {
      "name": "private-subnet",
      "enable_dhcp": true,
      "network_id": "db193ab3-96e3-4cb3-8fc5-05f4296d0324",
      "segment_id": null,
      "project_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
      "tenant_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
      "dns_nameservers": [],
      "allocation_pools": [
        {
          "start": "10.0.0.2",
          "end": "10.0.0.254"
        }
      ],
      "host_routes": [],
      "ip_version": 4,
      "gateway_ip": "10.0.0.1",
      "cidr": "10.0.0.0/24",
      "id": "08eae331-0402-425a-923c-34f7cfe39c1b",
      "created_at": "2016-10-10T14:35:34Z",
      "description": "",
      "ipv6_address_mode": null,
      "ipv6_ra_mode": null,
      "revision_number": 2,
      "service_types": [],
      "subnetpool_id": null,
      "updated_at": "2016-10-10T14:35:34Z"
    },
    {
      "name": "my_subnet",
      "enable_dhcp": true,
      "network_id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
      "segment_id": null,
      "project_id": "4fd44f30292945e481c7b8a0c8908869",
      "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
      "dns_nameservers": [],
      "allocation_pools": [
        {
          "start": "192.0.0.2",
          "end": "192.255.255.254"
        }
      ],
      "host_routes": [],
      "ip_version": 4,
      "gateway_ip": "192.0.0.1",
      "cidr": "192.0.0.0/8",
      "id": "54d6f61d-db07-451c-9ab3-b9609b6b6f0b",
      "created_at": "2016-10-10T14:35:47Z",

```

```
    "description": "",
    "ipv6_address_mode": null,
    "ipv6_ra_mode": null,
    "revision_number": 2,
    "service_types": [],
    "subnetpool_id": null,
    "updated_at": "2016-10-10T14:35:47Z"
  }
}
```

**POST**

/v2.0/subnets

## Create subnet

Creates a subnet on a network.

OpenStack Networking does not try to derive the correct IP version from the CIDR. If you do not specify the `gateway_ip` attribute, OpenStack Networking allocates an address from the CIDR for the gateway for the subnet.

To specify a subnet without a gateway, set the `gateway_ip` attribute to `null` in the request body. If you do not specify the `allocation_pools` attribute, OpenStack Networking automatically allocates pools for covering all IP addresses in the CIDR, excluding the address reserved for the subnet gateway. Otherwise, you can explicitly specify allocation pools as shown in the following example.

When you specify both the `allocation_pools` and `gateway_ip` attributes, you must ensure that the gateway IP does not overlap with the allocation pools; otherwise, the call returns the `Conflict` (409) response code.

A subnet can have one or more name servers and host routes. Hosts in this subnet use the name servers. Devices with IP addresses from this subnet, not including the local subnet route, use the host routes.

Specify the `ipv6_ra_mode` and `ipv6_address_mode` attributes to create subnets that support IPv6 configurations, such as stateless address autoconfiguration (SLAAC), DHCPv6 stateful, and DHCPv6 stateless configurations.

A subnet can optionally be associated with a network segment when it is created by specifying the `segment_id` of a valid segment on the specified network. A network with subnets associated in this way is called a routed network. On any given network, all of the subnets must be associated with segments or none of them can be. Neutron enforces this invariant. Currently, routed networks are only supported for provider networks.

Normal response codes: 201

Error response codes: 400, 401, 403, 404, 409

### Request

Name	In	Type	Description
subnet	body	string	A subnet object.



Name	In	Type	Description
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
enable_dhcp (Optional)	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet. Default is true.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers (Optional)	body	array	List of dns name servers associated with the subnet. Default is an empty list.
allocation_pools (Optional)	body	array	Allocation pools with start and end IP addresses for this subnet. If allocation_pools are not specified, OpenStack Networking automatically allocates pools for covering all IP addresses in the CIDR, excluding the address reserved for the subnet gateway by default.
host_routes (Optional)	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters. Default value is an empty list.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip (Optional)	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet. If the gateway_ip is not specified, OpenStack Networking allocates an address from the CIDR for the gateway for the subnet by default.
cidr	body	string	The CIDR of the subnet.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
ipv6_address_mode (Optional)	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless.

Name	In	Type	Description
ipv6_ra_mode (Optional)	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless.
segment_id (Optional)	body	string	The ID of a network segment the subnet is associated with. It is available when segment extension is enabled.
subnetpool_id (Optional)	body	string	The ID of the subnet pool associated with the subnet.

## Request Example

```
{
  "subnet": {
    "network_id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
    "ip_version": 4,
    "cidr": "192.168.199.0/24"
  }
}
```

## Response Parameters

Name	In	Type	Description
subnet	body	string	A subnet object.
id	body	string	The ID of the subnet.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
enable_dhcp	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers	body	array	List of dns name servers associated with the subnet.
allocation_pools	body	array	Allocation pools with start and end IP addresses for this subnet.
host_routes	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip	body	string	Gateway IP of this subnet. If the value

Name	In	Type	Description
			is null that implies no gateway is associated with the subnet.
cidr	body	string	The CIDR of the subnet.
created_at	body	string	Time at which the subnet has been created.
description	body	string	A human-readable description for the resource.
ipv6_address_mode	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
ipv6_ra_mode	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
revision_number	body	string	The revision number of the subnet.
service_types	body	string	The service types associated with the subnet.
subnetpool_id	body	string	The ID of the subnet pool associated with the subnet.
segment_id	body	string	The ID of a network segment the subnet is associated with. It is available when segment extension is enabled.
updated_at	body	string	Time at which the subnet has been updated.

## Response Example

```
{
  "subnet": {
    "name": "",
    "enable_dhcp": true,
    "network_id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
    "segment_id": null,
    "project_id": "4fd44f30292945e481c7b8a0c8908869",
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "dns_nameservers": [],
    "allocation_pools": [
      {
        "start": "192.168.199.2",
        "end": "192.168.199.254"
      }
    ],
    "host_routes": [],
    "ip_version": 4,
    "gateway_ip": "192.168.199.1",
    "cidr": "192.168.199.0/24",
    "id": "3b80198d-4f7b-4f77-9ef5-774d54e17126",
  }
}
```

```

    "created_at": "2016-10-10T14:35:47Z",
    "description": "",
    "ipv6_address_mode": null,
    "ipv6_ra_mode": null,
    "revision_number": 2,
    "service_types": [],
    "subnetpool_id": null,
    "updated_at": "2016-10-10T14:35:47Z"
  }
}

```

**POST**

/v2.0/subnets

## Bulk create subnet

Creates multiple subnets in a single request. Specify a list of subnets in the request body.

The bulk create operation is always atomic. Either all or no subnets in the request body are created.

Normal response codes: 201

Error response codes: 400, 401, 403, 404, 409

## Request

Name	In	Type	Description
subnets	body	array	A list of subnet objects.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
enable_dhcp (Optional)	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet. Default is true.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers	body	array	List of dns name servers

Name	In	Type	Description
(Optional)			associated with the subnet. Default is an empty list.
allocation_pools (Optional)	body	array	Allocation pools with start and end IP addresses for this subnet. If allocation_pools are not specified, OpenStack Networking automatically allocates pools for covering all IP addresses in the CIDR, excluding the address reserved for the subnet gateway by default.
host_routes (Optional)	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters. Default value is an empty list.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip (Optional)	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet. If the gateway_ip is not specified, OpenStack Networking allocates an address from the CIDR for the gateway for the subnet by default.
cidr	body	string	The CIDR of the subnet.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
ipv6_address_mode (Optional)	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless.
ipv6_ra_mode (Optional)	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless.
segment_id (Optional)	body	string	The ID of a network segment the subnet is associated with. It is available when segment extension is enabled.
subnetpool_id	body	string	The ID of the subnet pool

Name	In	Type	Description
(Optional)			associated with the subnet.

## Request Example

```
{
  "subnets": [
    {
      "cidr": "192.168.199.0/24",
      "ip_version": 4,
      "network_id": "e6031bc2-901a-4c66-82da-f4c32ed89406"
    },
    {
      "cidr": "10.56.4.0/22",
      "ip_version": 4,
      "network_id": "64239a54-dcc4-4b39-920b-b37c2144effa"
    }
  ]
}
```

## Response Parameters

Name	In	Type	Description
subnets	body	array	A list of subnet objects.
id	body	string	The ID of the subnet.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
enable_dhcp	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers	body	array	List of dns name servers associated with the subnet.
allocation_pools	body	array	Allocation pools with start and end IP addresses for this subnet.
host_routes	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet.
cidr	body	string	The CIDR of the subnet.

Name	In	Type	Description
created_at	body	string	Time at which the subnet has been created.
description	body	string	A human-readable description for the resource.
ipv6_address_mode	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
ipv6_ra_mode	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
revision_number	body	string	The revision number of the subnet.
segment_id	body	string	The ID of a network segment the subnet is associated with. It is available when segment extension is enabled.
service_types	body	string	The service types associated with the subnet.
subnetpool_id	body	string	The ID of the subnet pool associated with the subnet.
updated_at	body	string	Time at which the subnet has been updated.

## Response Example

```
{
  "subnets": [
    {
      "allocation_pools": [
        {
          "end": "192.168.199.254",
          "start": "192.168.199.2"
        }
      ],
      "cidr": "192.168.199.0/24",
      "dns_nameservers": [],
      "enable_dhcp": true,
      "gateway_ip": "192.168.199.1",
      "host_routes": [],
      "id": "0468a7a7-290d-4127-aedd-6c9449775a24",
      "ip_version": 4,
      "name": "",
      "network_id": "e6031bc2-901a-4c66-82da-f4c32ed89406",
      "segment_id": null,
      "project_id": "d19231fc08ec4bc4829b668040d34512",
      "tenant_id": "d19231fc08ec4bc4829b668040d34512",
      "created_at": "2016-10-10T14:35:47Z",
      "description": "",
      "ipv6_address_mode": null,
      "ipv6_ra_mode": null,
      "revision_number": 2,
    }
  ]
}
```

```

    "service_types": [],
    "subnetpool_id": null,
    "updated_at": "2016-10-10T14:35:47Z"
  },
  {
    "allocation_pools": [
      {
        "end": "10.56.7.254",
        "start": "10.56.4.2"
      }
    ],
    "cidr": "10.56.4.0/22",
    "dns_nameservers": [],
    "enable_dhcp": true,
    "gateway_ip": "10.56.4.1",
    "host_routes": [],
    "id": "b0e7435c-1512-45fb-aa9e-9a7c5932fb30",
    "ip_version": 4,
    "name": "",
    "network_id": "64239a54-dcc4-4b39-920b-b37c2144effa",
    "segment_id": null,
    "project_id": "d19231fc08ec4bc4829b668040d34512",
    "tenant_id": "d19231fc08ec4bc4829b668040d34512",
    "created_at": "2016-10-10T14:35:34Z",
    "description": "",
    "ipv6_address_mode": null,
    "ipv6_ra_mode": null,
    "revision_number": 2,
    "service_types": [],
    "subnetpool_id": null,
    "updated_at": "2016-10-10T14:35:34Z"
  }
]
}

```

## GET

/v2.0/subnets/{subnet\_id}

## Show subnet details

Shows details for a subnet.

Use the fields query parameter to filter the results.

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
subnet_id	path	string	The ID of the subnet.

## Response Parameters

Name	In	Type	Description
subnet	body	string	A subnet object.
id	body	string	The ID of the subnet.



Name	In	Type	Description
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
created_at	body	string	Time at which the subnet has been created.
name	body	string	Human-readable name of the resource.
enable_dhcp	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers	body	array	List of dns name servers associated with the subnet.
allocation_pools	body	array	Allocation pools with start and end IP addresses for this subnet.
host_routes	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet.
cidr	body	string	The CIDR of the subnet.
updated_at	body	string	Time at which the subnet has been updated.
description	body	string	A human-readable description for the resource.
ipv6_address_mode	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
ipv6_ra_mode	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
revision_number	body	string	The revision number of the subnet.

Name	In	Type	Description
segment_id	body	string	The ID of a network segment the subnet is associated with. It is available when segment extension is enabled.
service_types	body	string	The service types associated with the subnet.
subnetpool_id	body	string	The ID of the subnet pool associated with the subnet.

## Response Example

```
{
  "subnet": {
    "name": "my_subnet",
    "enable_dhcp": true,
    "network_id": "d32019d3-bc6e-4319-9c1d-6722fc136a22",
    "segment_id": null,
    "project_id": "4fd44f30292945e481c7b8a0c8908869",
    "tenant_id": "4fd44f30292945e481c7b8a0c8908869",
    "created_at": "2016-03-08T20:19:41",
    "dns_nameservers": [],
    "allocation_pools": [
      {
        "start": "192.0.0.2",
        "end": "192.255.255.254"
      }
    ],
    "host_routes": [],
    "ip_version": 4,
    "gateway_ip": "192.0.0.1",
    "cidr": "192.0.0.0/8",
    "updated_at": "2016-03-08T20:19:41",
    "id": "54d6f61d-db07-451c-9ab3-b9609b6b6f0b",
    "description": "",
    "ipv6_address_mode": null,
    "ipv6_ra_mode": null,
    "revision_number": 2,
    "service_types": [],
    "subnetpool_id": null
  }
}
```

**PUT**

/v2.0/subnets/{subnet\_id}

## Update subnet

Updates a subnet.

Some attributes, such as IP version (ip\_version), CIDR (cidr), and segment (segment\_id) cannot be updated. Attempting to update these attributes results in a 400 Bad Request error.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
subnet_id	path	string	The ID of the subnet.
name (Optional)	body	string	Human-readable name of the resource.
enable_dhcp (Optional)	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet. Default is true.
dns_nameservers (Optional)	body	array	List of dns name servers associated with the subnet. Default is an empty list.
allocation_pools (Optional)	body	array	Allocation pools with start and end IP addresses for this subnet. If allocation_pools are not specified, OpenStack Networking automatically allocates pools for covering all IP addresses in the CIDR, excluding the address reserved for the subnet gateway by default.
host_routes (Optional)	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters. Default value is an empty list.
gateway_ip (Optional)	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet. If the gateway_ip is not specified, OpenStack Networking allocates an address from the CIDR for the gateway for the subnet by default.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "subnet": {
    "name": "my_subnet"
  }
}
```

## Response Parameters

Name	In	Type	Description
subnet	body	string	A subnet object.

Name	In	Type	Description
id	body	string	The ID of the subnet.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
enable_dhcp	body	boolean	Indicates whether dhcp is enabled or disabled for the subnet.
network_id	body	string	The ID of the network to which the subnet belongs.
dns_nameservers	body	array	List of dns name servers associated with the subnet.
allocation_pools	body	array	Allocation pools with start and end IP addresses for this subnet.
host_routes	body	array	Additional routes for the subnet. A list of dictionaries with destination and nexthop parameters.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
gateway_ip	body	string	Gateway IP of this subnet. If the value is null that implies no gateway is associated with the subnet.
cidr	body	string	The CIDR of the subnet.
created_at	body	string	Time at which the subnet has been created.
description	body	string	A human-readable description for the resource.
ipv6_address_mode	body	string	The IPv6 address modes specifies mechanisms for assigning IP addresses. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
ipv6_ra_mode	body	string	The IPv6 router advertisement specifies whether the networking service should transmit ICMPv6 packets, for a

Name	In	Type	Description
			subnet. Value is slaac, dhcpv6-stateful, dhcpv6-stateless or null.
revision_number	body	string	The revision number of the subnet.
segment_id	body	string	The ID of a network segment the subnet is associated with. It is available when segment extension is enabled.
service_types	body	string	The service types associated with the subnet.
subnetpool_id	body	string	The ID of the subnet pool associated with the subnet.
updated_at	body	string	Time at which the subnet has been updated.

## Response Example

```
{
  "subnet": {
    "name": "my_subnet",
    "enable_dhcp": true,
    "network_id": "db193ab3-96e3-4cb3-8fc5-05f4296d0324",
    "segment_id": null,
    "project_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
    "tenant_id": "26a7980765d0414dbc1fc1f88cdb7e6e",
    "dns_nameservers": [],
    "allocation_pools": [
      {
        "start": "10.0.0.2",
        "end": "10.0.0.254"
      }
    ],
    "host_routes": [],
    "ip_version": 4,
    "gateway_ip": "10.0.0.1",
    "cidr": "10.0.0.0/24",
    "id": "08eae331-0402-425a-923c-34f7cfe39c1b",
    "description": ""
  }
}
```

## DELETE

/v2.0/subnets/{subnet\_id}

## Delete subnet

Deletes a subnet.

The operation fails if subnet IP addresses are still allocated.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
subnet_id	path	string	The ID of the subnet.

## Response

There is no body content for the response of a successful DELETE request.

# Security

## FWaaS v1.0 (DEPRECATED) (fw, firewalls, firewall\_policies, firewall\_rules)

Use the Firewall-as-a-Service (FWaaS) v1.0 extension to deploy firewalls to protect your networks.

The FWaaS extension enables you to:

- Apply firewall rules on traffic entering and leaving project networks.
- Apply TCP, UDP, ICMP, or protocol-agnostic rules.
- Create and share firewall policies that hold an ordered collection of the firewall rules.
- Audit firewall rules and policies.

This extension introduces these resources:

- `firewall`. A logical firewall resource that a project can instantiate and manage. A firewall can have one firewall policy.
- `firewall_policy`. An ordered collection of firewall rules. You can share a firewall policy across projects. You can include a firewall policy as part of an audit workflow so that an authorized relevant entity can audit the firewall policy. This entity can differ from the user who created, or the projects that use, the firewall policy.
- `firewall_rule`. A collection of attributes, such as ports and IP addresses. These attributes define match criteria and an action to take, such as allow or deny, on matched data traffic.

**GET**

/v2.0/fw/firewall\_policies

### List firewall policies

Lists all firewall policies.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using fields parameter, the API returns only the requested set of attributes. fields parameter can be specified multiple times. For example, if you specify fields=id&fields=name in the request URL, only id and name attributes will be returned.

## Response Parameters

Name	In	Type	Description
tenant_id	body	string	The ID of the project.
firewall_policies	body	array	A list of firewall_policy objects.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable description for the resource.
firewall_rules	body	array	A list of the IDs for firewall rule associated with the firewall policy.
id	body	string	The ID of the policy that is associated with the firewall.
name	body	string	Human-readable name of the resource.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
project_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_policies": [
    {
      "audited": false,
      "description": "",
      "firewall_rules": [
        "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
      ],
      "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "name": "test-policy",
      "shared": false,
    }
  ]
}
```

```
"project_id": "45977fa2dbd7482098dd68d0d8970117",  
"tenant_id": "45977fa2dbd7482098dd68d0d8970117"
```

**POST**

/v2.0/fw/firewall\_policies

## Create firewall policy

Creates a firewall policy.

Normal response codes: 201

Error response codes: 400, 401

### Request

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.
firewall_rules_id (Optional)	body	array	A list of rules to associate with the firewall policy.
name	body	string	Human-readable name of the resource.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

### Request Example

```
{  
  "firewall_policy": {  
    "firewall_rules": [  
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"  
    ],  
    "name": "test-policy"  
  }  
}
```



}

## Response Parameters

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.
name	body	string	Human-readable name of the resource.
firewall_rules	body	array	A list of the IDs for firewall rule associated with the firewall policy.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
id	body	string	The ID of the policy that is associated with the firewall.
description	body	string	A human-readable description for the resource.

**GET**

/v2.0/fw/firewall\_policies/{firewall\_policy\_id}

## Show firewall policy details

Shows details for a firewall policy.

If the user is not an administrative user and the firewall policy object does not belong to the project, this call returns the Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.

## Response Parameters

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.

Name	In	Type	Description
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to <code>false</code> . To audit the policy, explicitly set this attribute to <code>true</code> .
description	body	string	A human-readable description for the resource.
firewall_rules	body	array	A list of the IDs for firewall rule associated with the firewall policy.
id	body	string	The ID of the policy that is associated with the firewall.
name	body	string	Human-readable name of the resource.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.

## Response Example

```
{
  "firewall_policy": {
    "audited": false,
    "description": "",
    "firewall_rules": [
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ],
    "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "name": "test-policy",
    "shared": false,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**PUT**

/v2.0/fw/firewall\_policies/{firewall\_policy\_id}

## Update firewall policy

Updates a firewall policy.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.
firewall_rule	body	object	A firewall_rule object.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
name	body	string	Human-readable name of the resource.

## Request Example

```
{
  "firewall_policy": {
    "firewall_rules": [
      "a08ef905-0ff6-4784-8374-175fffe7dade",
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ]
  }
}
```

## Response Parameters

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.
project_id	body	string	The ID of the project.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable description for the resource.
firewall_rules	body	array	A list of the IDs for firewall rule associated with the firewall policy.
id	body	string	The ID of the policy that is

Name	In	Type	Description
			associated with the firewall.
name	body	string	Human-readable name of the resource.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_policy": {
    "audited": false,
    "description": "",
    "firewall_rules": [
      "a08ef905-0ff6-4784-8374-175fffe7dade",
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ],
    "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "name": "test-policy",
    "shared": false,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

## DELETE

/v2.0/fw/firewall\_policies/{firewall\_policy\_id}

## Delete firewall policy

Deletes a firewall policy.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.

## Response

There is no body content for the response of a successful DELETE request.

## PUT

/v2.0/fw/firewall\_policies/{firewall\_policy\_id}/insert\_rule

## Insert rule into a firewall policy

Insert firewall rule into a policy.

A `firewall_rule_id` is inserted relative to the position of the `firewall_rule_id` set in `insert_before` or `insert_after`. If `insert_before` is set, `insert_after` is ignored. If both `insert_before` and `insert_after` are not set, the new `firewall_rule_id` is inserted at the top of the policy.

Normal response codes: 200

Error response codes: 400, 401, 404, 409

### Request

Name	In	Type	Description
<code>firewall_policy_id</code>	path	string	The ID of the firewall policy.
<code>firewall_rule_id</code>	body	string	The ID of the firewall rule.
<code>insert_after</code> (Optional)	body	string	The ID of the <code>firewall_rule</code> . A new <code>firewall_rule</code> will be inserted after this <code>firewall_rule</code> .
<code>insert_before</code> (Optional)	body	string	The ID of the <code>firewall_rule</code> . A new <code>firewall_rule</code> will be inserted before this <code>firewall_rule</code> .

### Request Example

```
{
  "firewall_rule_id": "7bc34b8c-8d3b-4ada-a9c8-1f4c11c65692",
  "insert_after": "a08ef905-0ff6-4784-8374-175fffe7dade",
  "insert_before": ""
}
```

### Response Parameters

Name	In	Type	Description
<code>audited</code>	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
<code>description</code>	body	string	A human-readable description for the resource.
<code>firewall_list</code>	body	array	A list of the IDs of firewalls associated with the firewall policy.
<code>firewall_rules</code>	body	array	A list of the IDs for firewall rule associated with the firewall policy.
<code>tenant_id</code>	body	string	The ID of the project.

Name	In	Type	Description
project_id	body	string	The ID of the project.
id	body	string	The ID of the policy that is associated with the firewall.
name	body	string	Human-readable name of the resource.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.

## Response Example

```
{
  "audited": false,
  "description": "",
  "firewall_list": [],
  "firewall_rules": [
    "a08ef905-0ff6-4784-8374-175fffe7dade",
    "7bc34b8c-8d3b-4ada-a9c8-1f4c11c65692",
    "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
  ],
  "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
  "name": "test-policy",
  "shared": false,
  "project_id": "45977fa2dbd7482098dd68d0d8970117",
  "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
}
```

**PUT**

/v2.0/fw/firewall\_policies/{firewall\_policy\_id}/remove\_rule

## Remove rule from firewall policy

Remove firewall rule from a policy.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.
firewall_rule_id	body	string	The ID of the firewall rule.

## Request Example

```
{
  "firewall_rule_id": "7bc34b8c-8d3b-4ada-a9c8-1f4c11c65692"
}
```

## Response Parameters

Name	In	Type	Description
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to <code>false</code> . To audit the policy, explicitly set this attribute to <code>true</code> .
description	body	string	A human-readable description for the resource.
firewall_list	body	array	A list of the IDs of firewalls associated with the firewall policy.
firewall_rules	body	array	A list of the IDs for firewall rule associated with the firewall policy.
id	body	string	The ID of the FWaaS v1 firewall.
name	body	string	Human-readable name of the resource.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.

## Response Example

```
{
  "audited": false,
  "description": "",
  "firewall_list": [],
  "firewall_rules": [
    "a08ef905-0ff6-4784-8374-175fffe7dade",
    "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
  ],
  "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
  "name": "test-policy",
  "shared": false,
  "project_id": "45977fa2dbd7482098dd68d0d8970117",
  "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
}
```

**GET**

/v2.0/fw/firewall\_rules

### List firewall rules

Lists all firewall rules.

The list might be empty.

Normal response codes: 200

Error response codes: 401, 403

Request

Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid value is allow, deny or reject. Default is deny.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR. No default.
destination_port	body	string	The destination port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
id	body	string	The ID of the FWaaS v1 firewall.
ip_version	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
name	body	string	Human-readable name of the resource.
position	body	integer	Read-only attribute that the API assigns to this rule when it associates it with a firewall



Name	In	Type	Description
			policy. This value indicates the position of this rule in that firewall policy. This position number starts at 1. If the firewall rule is not associated with any policy, the position is null.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR.
source_port (Optional)	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.

## Response Example

```
{
  "firewall_rules": [
    {
      "action": "allow",
      "description": "",
      "destination_ip_address": null,
      "destination_port": "80",
      "enabled": true,
      "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
      "ip_version": 4,
      "name": "ALLOW_HTTP",
      "position": 1,
      "protocol": "tcp",
      "shared": false,
      "source_ip_address": null,
      "source_port": null,
      "project_id": "45977fa2dbd7482098dd68d0d8970117",
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

**POST**

/v2.0/fw/firewall\_rules

## Create firewall rule

Creates a firewall rule.

Normal response codes: 201

## Error response codes: 400, 401

### Request

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action (Optional)	body	string	The action that the API performs on traffic that matches the firewall rule. Valid value is allow or deny. Default is deny.
destination_port (Optional)	body	string	The destination port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled (Optional)	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
enabled (Optional)	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
name	body	string	Human-readable name of the resource.

Name	In	Type	Description
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
destination_ip_addresses (Optional)	body	string	The destination IPv4 or IPv6 address or CIDR. No default.
source_port	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.

### Request Example

```
{
  "firewall_rule": {
    "action": "allow",
    "destination_port": "80",
    "enabled": true,
    "name": "ALLOW_HTTP",
    "protocol": "tcp"
  }
}
```

### Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid value is allow, deny or reject. Default is deny.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR. No default.

Name	In	Type	Description
destination_port	body	string	The destination port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
id	body	string	The ID of the FWaaS v1 firewall.
ip_version	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
name	body	string	Human-readable name of the resource.
position	body	integer	Read-only attribute that the API assigns to this rule when it associates it with a firewall policy. This value indicates the position of this rule in that firewall policy. This position number starts at 1. If the firewall rule is not associated with any policy, the position is null.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR.
source_port (Optional)	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.

## Response Example

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_ip_address": null,
    "destination_port": "80",
    "enabled": true,
    "firewall_policy_id": null,
    "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
    "ip_version": 4,
    "name": "ALLOW_HTTP",
    "position": null,
    "protocol": "tcp",
    "shared": false,
    "source_ip_address": null,
    "source_port": null,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**GET**

/v2.0/fw/firewall\_rules/{firewall\_rule\_id}

## Show firewall rule details

Shows details for a firewall rule.

If the user is not an administrative user and the firewall rule object does not belong to the project, this call returns the Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
firewall_rule_id	path	string	The ID for the firewall rule.

## Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid value is allow, deny or reject. Default is deny.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.

Name	In	Type	Description
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR. No default.
destination_port	body	string	The destination port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
firewall_policy_id (Optional)	body	string	Read-only attribute that the API populates with the ID of the firewall policy when you associate this firewall rule with a policy. You can associate a firewall rule with one policy at a time. You can update this association can to a different firewall policy. If you do not associate the rule with any policy, this attribute is null.
id	body	string	The ID of the firewall rule.
ip_version	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
name	body	string	Human-readable name of the resource.
position	body	integer	Read-only attribute that the API assigns to this rule when it associates it with a firewall policy. This value indicates the position of this rule in that firewall policy. This position number starts at 1. If the firewall rule is not associated with any policy, the position is null.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR.
source_port (Optional)	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.

## Response Example

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_ip_address": null,
    "destination_port": "80",
    "enabled": true,
    "firewall_policy_id": null,
    "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
    "ip_version": 4,
    "name": "ALLOW_HTTP",
    "position": null,
    "protocol": "tcp",
    "shared": false,
    "source_ip_address": null,
    "source_port": null,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**PUT**

/v2.0/fw/firewall\_rules/{firewall\_rule\_id}

## Update firewall rule

Updates a firewall rule.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
firewall_rule_id	path	string	The ID for the firewall rule.
firewall_rule	body	object	A firewall_rule object.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through

Name	In	Type	Description
			authorization policies.
enabled (Optional)	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
destination_ip_addresses (Optional)	body	string	The destination IPv4 or IPv6 address or CIDR. No default.
source_port	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
action (Optional)	body	string	The action that the API performs on traffic that matches the firewall rule. Valid value is allow or deny. Default is deny.
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
destination_port (Optional)	body	string	The destination port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
name	body	string	Human-readable name of the resource.

### Request Example

```
{
  "firewall_rule": {
    "shared": "true"
  }
}
```

### Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid value is allow, deny or reject. Default is deny.
description	body	string	A human-readable description for the resource.



Name	In	Type	Description
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
enabled (Optional)	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
source_port	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
ip_version (Optional)	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR. No default.
destination_port	body	string	The destination port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid value is true or false. Default is true.
firewall_policy_id (Optional)	body	string	Read-only attribute that the API populates with the ID of the firewall policy when you associate this firewall rule with a policy. You can associate a firewall rule with one policy at a time. You can update this association can to a different firewall policy. If you do not associate the rule with any policy, this attribute is null.
id	body	string	The ID of the firewall rule.
ip_version	body	integer	The IP protocol version. Valid value is 4 or 6. Default is 4.
name	body	string	Human-readable name of the resource.
position	body	integer	Read-only attribute that the API assigns to

Name	In	Type	Description
			this rule when it associates it with a firewall policy. This value indicates the position of this rule in that firewall policy. This position number starts at 1. If the firewall rule is not associated with any policy, the position is null.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
shared	body	boolean	Admin-only. Indicates whether this network is shared across all tenants.
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR.
source_port (Optional)	body	string	The source port or port range. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.

## Response Example

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_ip_address": null,
    "destination_port": "80",
    "enabled": true,
    "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
    "ip_version": 4,
    "name": "ALLOW_HTTP",
    "position": 1,
    "protocol": "tcp",
    "shared": true,
    "source_ip_address": null,
    "source_port": null,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

## DELETE

/v2.0/fw/firewall\_rules/{firewall\_rule\_id}

## Delete firewall rule

Deletes a firewall rule.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
firewall_rule_id	path	string	The ID for the firewall rule.

## Response

There is no body content for the response of a successful DELETE request.

**GET**

/v2.0/fw/firewalls

## List firewalls

Lists all firewalls.

The list might be empty.

Normal response codes: 200

Error response codes: 401, 403

## Request

### Response Parameters

Name	In	Type	Description
firewalls	body	array	A list of firewall_rule objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
description	body	string	A human-readable description for the resource.
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
id	body	string	The ID of the FWaaS v1 firewall.
name	body	string	Human-readable name of the resource.
status	body	string	The status of the firewall service. Values are ACTIVE, INACTIVE, ERROR, DOWN, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.

## Response Example

```
{
  "firewalls": [
    {
      "admin_state_up": true,
      "description": "",
      "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
      "name": "",
      "status": "ACTIVE",
      "router_ids": [
        "650bfd2f-7766-4a0d-839f-218f33e16998"
      ],
      "project_id": "45977fa2dbd7482098dd68d0d8970117",
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

**POST**

/v2.0/fw/firewalls

## Create firewall

Creates a firewall.

The firewall must be associated with a firewall policy.

If admin\_state\_up is false, the firewall would block all traffic.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
firewall	body	object	A firewall object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
name	body	string	Human-readable name of the resource.
router_ids (Optional)	body	array	A list of IDs for routers that are associated with the firewall.

## Request Example

```
{
  "firewall": {
    "admin_state_up": true,
    "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c"
  }
}
```

## Response Parameters

Name	In	Type	Description
firewall	body	object	A firewall object.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
id	body	string	The ID of the FWaaS v1 firewall.
name	body	string	Human-readable name of the resource.
status	body	string	The status of the firewall service. Values are ACTIVE, INACTIVE, ERROR, DOWN, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
router_ids	body	array	A list of IDs for routers that are associated with the firewall.

## Response Example

```
{
  "firewall": {
    "admin_state_up": true,
    "description": "",
    "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
    "name": "",
    "status": "PENDING_CREATE",
    "router_ids": [
      "650bfd2f-7766-4a0d-839f-218f33e16998"
    ],
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

GET

/v2.0/fw/firewalls/{firewall\_id}

## Show firewall details

Shows details for a firewall.

If the user is not an administrative user and the firewall object does not belong to the project, this call returns the Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
firewall_id	path	string	The ID of the firewall.

### Response Parameters

Name	In	Type	Description
firewall	body	object	A firewall object.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
status	body	string	The status of the firewall service. Values are ACTIVE, INACTIVE, ERROR, DOWN, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
firewall_policy_id (Optional)	body	string	Read-only attribute that the API populates with the ID of the firewall policy when you associate this firewall rule with a policy. You can associate a firewall rule with one policy at a time. You can update this association can to a different firewall policy. If you do not associate the rule with any policy, this attribute is null.
id	body	string	The ID of the firewall rule.
name	body	string	Human-readable name of the resource.
router_ids	body	array	A list of IDs for routers that are associated with the firewall.

## Response Example

```
{
  "firewall": {
    "admin_state_up": true,
    "description": "",
    "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
    "name": "",
    "status": "ACTIVE",
    "router_ids": [
      "650bfd2f-7766-4a0d-839f-218f33e16998"
    ],
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**PUT**

/v2.0/fw/firewalls/{firewall\_id}

## Update firewall

Updates a firewall.

To update a service, the service status cannot be a PENDING\_\* status.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
firewall_id	path	string	The ID of the firewall.
firewall	body	object	A firewall object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
name	body	string	Human-readable name of the resource.
router_ids (Optional)	body	array	A list of IDs for routers that are associated with the firewall.

## Request Example

```
{
  "firewall": {
    "admin_state_up": "false"
  }
}
```

## Response Parameters

Name	In	Type	Description
firewall	body	object	A firewall object.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
status	body	string	The status of the firewall service. Values are ACTIVE, INACTIVE, ERROR, DOWN, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
firewall_policy_id	body	string	The ID of the policy that is associated with the firewall.
id	body	string	The ID of the FWaaS v1 firewall.
name	body	string	Human-readable name of the resource.
router_ids	body	array	A list of IDs for routers that are associated with the firewall.

## Response Example

```
{
  "firewall": {
    "admin_state_up": false,
    "description": "",
    "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
    "name": "",
    "status": "PENDING_UPDATE",
    "router_ids": [
      "650bfd2f-7766-4a0d-839f-218f33e16998"
    ],
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

### DELETE

/v2.0/fw/firewalls/{firewall\_id}

## Delete firewall

Deletes a firewall.

Normal response codes: 204

Error response codes: 401, 404, 409



## Request

Name	In	Type	Description
firewall_id	path	string	The ID of the firewall.

## Response

There is no body content for the response of a successful DELETE request.

## FWaaS v2.0 (CURRENT) (fwaas, firewall\_groups, firewall\_policies, firewall\_rules)

Use the Firewall-as-a-Service (FWaaS) v2.0 extension to deploy firewall groups to protect your networks.

The FWaaS extension enables you to:

- Apply firewall rules on traffic entering and leaving project networks.
- Apply TCP, UDP, ICMP, or protocol-agnostic rules.
- Create and share firewall policies that hold an ordered collection of firewall rules.
- Audit firewall rules and policies.

This extension introduces the following resources:

- `firewall_group`. A logical firewall resource that a project can create and manage. A firewall group can have a firewall policy for ingress traffic and/or a firewall policy for egress traffic.
- `firewall_policy`. An ordered collection of firewall rules. You can share a firewall policy across projects. You can include a firewall policy as part of an audit workflow so that an authorized relevant entity can audit the firewall policy. This entity can differ from the user who created, or the projects that use, the firewall policy.
- `firewall_rule`. A collection of attributes, such as source and destination ports, source and destination IP addresses, protocol, and IP version. These attributes define match criteria and an action to take, such as allow, reject, or deny, on matched data traffic.

**GET**

/v2.0/fwaas/firewall\_groups

### List firewall groups

Lists all firewall groups.

The list might be empty.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

## Error response codes: 401, 403

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
firewall_groups	body	array	A list of <code>firewall_group</code> objects.
admin_state_up	body	boolean	The administrative state of the firewall group, which is <code>up</code> ( <code>true</code> ) or <code>down</code> ( <code>false</code> ). Default is <code>true</code> .
description	body	object	A human-readable description of the firewall group.
egress_firewall_policy_id	body	string	The ID of the egress firewall policy for the firewall group.
id	body	string	The ID of the firewall group.
ingress_firewall_policy_id	body	string	The ID of the ingress firewall policy for the firewall group.
name	body	string	A human-readable name for the firewall group.
ports	body	array	A list of the IDs of the ports associated with the firewall group.
project_id	body	string	The ID of the project.
public	body	boolean	Indicates whether this firewall group is shared across all projects.
status	body	string	The status of the firewall group. Valid values are <code>ACTIVE</code> , <code>INACTIVE</code> , <code>ERROR</code> , <code>PENDING_UPDATE</code> , or <code>PENDING_DELETE</code> .
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_groups": [
    {
      "admin_state_up": true,
      "description": "",
      "egress_firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
      "ingress_firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "name": "",
      "ports": [
        "650bfd2f-7766-4a0d-839f-218f33e16998"
      ],
      "project_id": "45977fa2dbd7482098dd68d0d8970117",
      "status": "ACTIVE",
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

**GET**

/v2.0/fwaas/firewall\_groups/{firewall\_group\_id}

## Show firewall group details

Shows details for a firewall group.

If the user is not an administrative user and the firewall group object does not belong to the project, this call returns the FirewallGroupNotFound (404) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
firewall_group_id	path	string	The ID of the firewall group.

## Response Parameters

Name	In	Type	Description
firewall_group	body	object	A firewall_group object.
admin_state_up	body	boolean	The administrative state of the firewall group, which is up (true) or down (false). Default is true.
description	body	object	A human-readable description of the firewall group.
egress_firewall_policy_id	body	string	The ID of the egress firewall policy for the firewall group.
id	body	string	The ID of the firewall group.

Name	In	Type	Description
ingress_firewall_policy_id	body	string	The ID of the ingress firewall policy for the firewall group.
name	body	string	A human-readable name for the firewall group.
ports	body	array	A list of the IDs of the ports associated with the firewall group.
project_id	body	string	The ID of the project.
public	body	boolean	Indicates whether this firewall group is shared across all projects.
status	body	string	The status of the firewall group. Valid values are ACTIVE, INACTIVE, ERROR, PENDING_UPDATE, or PENDING_DELETE.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_group": {
    "admin_state_up": true,
    "description": "",
    "egress_firewall_policy_id": null,
    "id": "07411bda-0147-418b-af05-c8665630d937",
    "ingress_firewall_policy_id": null,
    "name": "",
    "project_id": "96108b04417b416e9b9bc788c11c42c9",
    "public": false,
    "status": "INACTIVE",
    "tenant_id": "96108b04417b416e9b9bc788c11c42c9"
  }
}
```

**POST**

/v2.0/fwaas/firewall\_groups

## Create firewall group

Creates a firewall group.

The firewall group may be associated with an ingress firewall policy and/or an egress firewall policy.

If admin\_state\_up is false, the firewall group will block all traffic.

Normal response codes: 201

## Error response codes: 400, 401

### Request

Name	In	Type	Description
firewall_group	body	object	A firewall_group object.
admin_state_up (Optional)	body	boolean	The administrative state of the firewall group, which is up (true) or down (false). Default is true.
description (Optional)	body	object	A human-readable description of the firewall group.
egress_firewall_policy_id (Optional)	body	string	The ID of the egress firewall policy for the firewall group.
ingress_firewall_policy_id (Optional)	body	string	The ID of the ingress firewall policy for the firewall group.
name (Optional)	body	string	A human-readable name for the firewall group.
ports (Optional)	body	array	A list of the IDs of the ports associated with the firewall group.
project_id (Optional)	body	string	The ID of the project.
public (Optional)	body	boolean	Indicates whether this firewall group is shared across all projects.
status (Optional)	body	string	The status of the firewall group. Valid values are ACTIVE, INACTIVE, ERROR, PENDING_UPDATE, or PENDING_DELETE.
tenant_id (Optional)	body	string	The ID of the project.

### Request Example

```
{
  "firewall_group": {
    "admin_state_up": false,
    "egress_firewall_policy_id": "14c9d3c1-b472-44f9-8226-30dc4ffd454c",
    "ingress_firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c"
  }
}
```

### Response Parameters

Name	In	Type	Description
firewall_group	body	object	A firewall_group object.

Name	In	Type	Description
admin_state_up	body	boolean	The administrative state of the firewall group, which is up (true) or down (false). Default is true.
description	body	object	A human-readable description of the firewall group.
egress_firewall_policy_id	body	string	The ID of the egress firewall policy for the firewall group.
id	body	string	The ID of the firewall group.
ingress_firewall_policy_id	body	string	The ID of the ingress firewall policy for the firewall group.
name	body	string	A human-readable name for the firewall group.
ports	body	array	A list of the IDs of the ports associated with the firewall group.
project_id	body	string	The ID of the project.
public	body	boolean	Indicates whether this firewall group is shared across all projects.
status	body	string	The status of the firewall group. Valid values are ACTIVE, INACTIVE, ERROR, PENDING_UPDATE, or PENDING_DELETE.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_group": {
    "admin_state_up": true,
    "description": "",
    "egress_firewall_policy_id": "1244ed87-b472-44f9-8226-30dc4ffd454c",
    "ingress_firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
    "name": "",
    "ports": [
      "650bfd2f-7766-4a0d-839f-218f33e16998"
    ],
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "public": true,
    "status": "PENDING_CREATE",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

PUT

/v2.0/fwaas/firewall\_groups/{firewall\_group\_id}

## Update firewall group

Updates a firewall group.

The firewall group cannot be updated if its status is a PENDING\_\* status.

Normal response codes: 200

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
firewall_group_id	path	string	The ID of the firewall group.
firewall_group	body	object	A firewall_group object.
admin_state_up (Optional)	body	boolean	The administrative state of the firewall group, which is up (true) or down (false). Default is true.
description (Optional)	body	object	A human-readable description of the firewall group.
egress_firewall_policy_id (Optional)	body	string	The ID of the egress firewall policy for the firewall group.
ingress_firewall_policy_id (Optional)	body	string	The ID of the ingress firewall policy for the firewall group.
name (Optional)	body	string	A human-readable name for the firewall group.
ports (Optional)	body	array	A list of the IDs of the ports associated with the firewall group.
status (Optional)	body	string	The status of the firewall group. Valid values are ACTIVE, INACTIVE, ERROR, PENDING_UPDATE, or PENDING_DELETE.

### Request Example

```
{
  "firewall_group": {
    "admin_state_up": "false"
  }
}
```

## Response Parameters

Name	In	Type	Description
firewall_group	body	object	A firewall_group object.
admin_state_up	body	boolean	The administrative state of the firewall group, which is up (true) or down (false). Default is true.
description	body	object	A human-readable description of the firewall group.
egress_firewall_policy_id	body	string	The ID of the egress firewall policy for the firewall group.
id	body	string	The ID of the firewall group.
ingress_firewall_policy_id	body	string	The ID of the ingress firewall policy for the firewall group.
name	body	string	A human-readable name for the firewall group.
ports	body	array	A list of the IDs of the ports associated with the firewall group.
project_id	body	string	The ID of the project.
public	body	boolean	Indicates whether this firewall group is shared across all projects.
status	body	string	The status of the firewall group. Valid values are ACTIVE, INACTIVE, ERROR, PENDING_UPDATE, or PENDING_DELETE.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_group": {
    "admin_state_up": false,
    "description": "",
    "egress_firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "ingress_firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "3b0ef8f4-82c7-44d4-a4fb-6177f9a21977",
    "name": "",
    "ports": [
      "650bfd2f-7766-4a0d-839f-218f33e16998"
    ],
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "status": "PENDING_UPDATE",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```



**DELETE**

/v2.0/fwaas/firewall\_groups/{firewall\_group\_id}

**Delete firewall group**

Deletes a firewall group.

Normal response codes: 204

Error response codes: 401, 404, 409

**Request**

Name	In	Type	Description
firewall_group_id	path	string	The ID of the firewall group.

**Response**

There is no body content for the response of a successful DELETE request.

**GET**

/v2.0/fwaas/firewall\_policies

**List firewall policies**

Lists all firewall policies.

The list might be empty.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

**Request**

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
firewall_policies	body	array	A list of firewall_policy objects.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable name of the firewall policy.
id	body	string	The ID of the firewall policy.
firewall_rules	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
name	body	string	A human-readable name of the firewall policy.
project_id	body	string	The ID of the project.
public	body	boolean	Set to true to make this firewall policy visible to other projects. Default is false.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_policies": [
    {
      "audited": false,
      "description": "",
      "firewall_rules": [
        "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
      ],
      "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "name": "test-policy",
      "project_id": "45977fa2dbd7482098dd68d0d8970117",
      "public": false,
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

**GET**

/v2.0/fwaas/firewall\_policies/{firewall\_policy\_id}

### Show firewall policy details

Shows details of a firewall policy.

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.

### Response Parameters

Name	In	Type	Description
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to <code>false</code> . To audit the policy, explicitly set this attribute to <code>true</code> .
description	body	string	A human-readable name of the firewall policy.
firewall_rules	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
id	body	string	The ID of the firewall policy.
name	body	string	A human-readable name of the firewall policy.
project_id	body	string	The ID of the project.
public	body	boolean	Set to <code>true</code> to make this firewall policy visible to other projects. Default is <code>false</code> .
tenant_id	body	string	The ID of the project.

### Response Example

```
{
  "firewall_policy": {
    "audited": false,
    "description": "",
    "firewall_rules": [
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ],
    "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "name": "test-policy",
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "public": false,
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**POST**

/v2.0/fwaas/firewall\_policies

## Create firewall policy

Creates a firewall policy.

Normal response codes: 201

Error response codes: 400, 401

### Request

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.
audited (Optional)	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description (Optional)	body	string	A human-readable name of the firewall policy.
firewall_rules (Optional)	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
name (Optional)	body	string	A human-readable name of the firewall policy.
project_id (Optional)	body	string	The ID of the project.
public (Optional)	body	boolean	Set to true to make this firewall policy visible to other projects. Default is false.
tenant_id (Optional)	body	string	The ID of the project.

### Request Example

```
{
  "firewall_policy": {
    "name": "test-policy",
    "firewall_rules": [
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ]
  }
}
```

### Response Parameters

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this

Name	In	Type	Description
			attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable name of the firewall policy.
firewall_rules	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
id	body	string	The ID of the firewall policy.
name	body	string	A human-readable name of the firewall policy.
project_id	body	string	The ID of the project.
public	body	boolean	Set to true to make this firewall policy visible to other projects. Default is false.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_policy": {
    "audited": false,
    "description": "",
    "firewall_rules": [
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ],
    "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "name": "test-policy",
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "public": false,
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**PUT**

/v2.0/fwaas/firewall\_policies/{firewall\_policy\_id}

## Update firewall policy

Updates a firewall policy.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.
firewall_policy	body	object	A firewall_policy object.
audited (Optional)	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description (Optional)	body	string	A human-readable name of the firewall policy.
firewall_rules (Optional)	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
name (Optional)	body	string	A human-readable name of the firewall policy.
project_id (Optional)	body	string	The ID of the project.
public (Optional)	body	boolean	Set to true to make this firewall policy visible to other projects. Default is false.
tenant_id (Optional)	body	string	The ID of the project.

## Request Example

```
{
  "firewall_policy": {
    "firewall_rules": [
      "a08ef905-0ff6-4784-8374-175fffe7dade",
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ]
  }
}
```

## Response Parameters

Name	In	Type	Description
firewall_policy	body	object	A firewall_policy object.
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable name of the firewall policy.

Name	In	Type	Description
firewall_rules	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
id	body	string	The ID of the firewall policy.
name	body	string	A human-readable name of the firewall policy.
public	body	boolean	Set to true to make this firewall policy visible to other projects. Default is false.
project_id	body	string	The ID of the project.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_policy": {
    "audited": false,
    "description": "",
    "firewall_rules": [
      "a08ef905-0ff6-4784-8374-175fffe7dade",
      "8722e0e0-9cc9-4490-9660-8c9a5732fbb0"
    ],
    "id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "name": "test-policy",
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "public": false,
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

## DELETE

/v2.0/fwaas/firewall\_policies/{firewall\_policy\_id}

## Delete firewall policy

Deletes a firewall policy.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/fwaas/firewall\_rules

## List firewall rules

Lists all firewall rules.

The list might be empty.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
firewall_rules	body	object	A list of <code>firewall_rule</code> objects.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid values are <code>allow</code> or <code>deny</code> . Default is <code>deny</code> .
description	body	string	A human-readable description of the firewall rule.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR for the firewall rule. No default.
destination_port	body	string	The destination port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a <code>:</code> separated range. For a port range, include both ends of the range. For example, <code>80:90</code> .
enabled	body	boolean	Set to <code>false</code> to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from



Name	In	Type	Description
			the firewall policy. Valid values are true or false. Default is true.
firewall_policy_id	body	string	The ID of the firewall policy.
id	body	string	The ID of the firewall rule.
ip_version	body	integer	The IP protocol version for the firewall rule. Valid values are 4 or 6. Default is 4.
name	body	string	A human-readable name of the firewall rule.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol for the firewall rule. Possible values are icmp, tcp, udp, or null.
public	body	boolean	Indicates whether this firewall rule is shared across all projects.
source_ip_address	body	string	The source IPv4 or IPv6 address or CIDR for the firewall rule. No default.
source_port	body	string	The source port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_rules": [
    {
      "action": "allow",
      "description": "",
      "destination_ip_address": null,
      "destination_port": "80",
      "enabled": true,
      "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
      "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
      "ip_version": 4,
      "name": "ALLOW_HTTP",
      "position": 1,
      "project_id": "45977fa2dbd7482098dd68d0d8970117",
      "protocol": "tcp",
      "public": false,
      "source_ip_address": null,
      "source_port": null,
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

**GET**  
/v2.0/fwaas/firewall\_rules/{firewall\_rule\_id}

## Show firewall rule details

Shows details for a firewall rule.

If the user is not an administrative user and the firewall rule object does not belong to the project, this call returns the Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
firewall_rule_id	path	string	The ID for the firewall rule.

### Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid values are allow or deny. Default is deny.
description	body	string	A human-readable description of the firewall rule.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR for the firewall rule. No default.
destination_port	body	string	The destination port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid values are true or false. Default is true.
firewall_policy_id	body	string	The ID of the firewall policy.
id	body	string	The ID of the firewall rule.
ip_version	body	integer	The IP protocol version for the firewall rule. Valid values are 4 or 6. Default is 4.
name	body	string	A human-readable name of the firewall rule.

Name	In	Type	Description
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol for the firewall rule. Possible values are icmp, tcp, udp, or null.
public	body	boolean	Indicates whether this firewall rule is shared across all projects.
source_ip_address	body	string	The source IPv4 or IPv6 address or CIDR for the firewall rule. No default.
source_port	body	string	The source port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_ip_address": null,
    "destination_port": "80",
    "enabled": true,
    "firewall_policy_id": null,
    "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
    "ip_version": 4,
    "name": "ALLOW_HTTP",
    "position": null,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "protocol": "tcp",
    "public": false,
    "source_ip_address": null,
    "source_port": null,
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

**POST**

/v2.0/fwaas/firewall\_rules

## Create firewall rule

Creates a firewall rule.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action (Optional)	body	string	The action that the API performs on traffic that matches the firewall rule. Valid values are allow or deny. Default is deny.
description (Optional)	body	string	A human-readable description of the firewall rule.
destination_ip_addresses (Optional)	body	string	The destination IPv4 or IPv6 address or CIDR for the firewall rule. No default.
destination_port (Optional)	body	string	The destination port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled (Optional)	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid values are true or false. Default is true.
ip_version (Optional)	body	integer	The IP protocol version for the firewall rule. Valid values are 4 or 6. Default is 4.
name (Optional)	body	string	A human-readable name of the firewall rule.
project_id (Optional)	body	string	The ID of the project.
protocol (Optional)	body	string	The IP protocol for the firewall rule. Possible values are icmp, tcp, udp, or null.
public (Optional)	body	boolean	Indicates whether this firewall rule is shared across all projects.
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR for the firewall rule. No default.
source_port (Optional)	body	string	The source port or port range for the firewall rule. A valid value is a port number, as an

Name	In	Type	Description
			integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
tenant_id (Optional)	body	string	The ID of the project.

### Request Example

```
{
  "firewall_rule": {
    "action": "allow",
    "destination_port": "80",
    "enabled": true,
    "name": "ALLOW_HTTP",
    "protocol": "tcp"
  }
}
```

### Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid values are allow or deny. Default is deny.
description	body	string	A human-readable description of the firewall rule.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR for the firewall rule. No default.
destination_port	body	string	The destination port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid values are true or false. Default is true.
firewall_policy_id	body	string	The ID of the firewall policy.

Name	In	Type	Description
id	body	string	The ID of the firewall rule.
ip_version	body	integer	The IP protocol version for the firewall rule. Valid values are 4 or 6. Default is 4.
name	body	string	A human-readable name of the firewall rule.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol for the firewall rule. Possible values are icmp, tcp, udp, or null.
public	body	boolean	Indicates whether this firewall rule is shared across all projects.
source_ip_address	body	string	The source IPv4 or IPv6 address or CIDR for the firewall rule. No default.
source_port	body	string	The source port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_rule": {
    "action": "deny",
    "description": "",
    "destination_ip_address": null,
    "destination_port": null,
    "enabled": true,
    "id": "1fd59b2f-cc87-435f-a244-1df2c0cc3f70",
    "ip_version": 4,
    "name": "rule3",
    "project_id": "95573613ec554b4b8df9f2679c64557b",
    "protocol": null,
    "public": false,
    "source_ip_address": null,
    "source_port": null,
    "tenant_id": "95573613ec554b4b8df9f2679c64557b"
  }
}
```

**PUT**

/v2.0/fwaas/firewall\_rules/{firewall\_rule\_id}

## Update firewall rule

Updates a firewall rule.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
firewall_rule_id	path	string	The ID for the firewall rule.
firewall_rule	body	object	A firewall_rule object.
action (Optional)	body	string	The action that the API performs on traffic that matches the firewall rule. Valid values are allow or deny. Default is deny.
description (Optional)	body	string	A human-readable description of the firewall rule.
destination_ip_addresses (Optional)	body	string	The destination IPv4 or IPv6 address or CIDR for the firewall rule. No default.
destination_port (Optional)	body	string	The destination port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled (Optional)	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning off rules without having to disassociate the rule from the firewall policy. Valid values are true or false. Default is true.
firewall_policy_id	body	string	The ID of the firewall policy.
ip_version (Optional)	body	integer	The IP protocol version for the firewall rule. Valid values are 4 or 6. Default is 4.
name (Optional)	body	string	A human-readable name of the firewall rule.
project_id (Optional)	body	string	The ID of the project.
protocol (Optional)	body	string	The IP protocol for the firewall rule. Possible values are icmp, tcp, udp, or null.

Name	In	Type	Description
public (Optional)	body	boolean	Indicates whether this firewall rule is shared across all projects.
source_ip_address (Optional)	body	string	The source IPv4 or IPv6 address or CIDR for the firewall rule. No default.
source_port (Optional)	body	string	The source port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
tenant_id (Optional)	body	string	The ID of the project.

### Request Example

```
{
  "firewall_rule": {
    "public": "true"
  }
}
```

### Response Parameters

Name	In	Type	Description
firewall_rule	body	object	A firewall_rule object.
action	body	string	The action that the API performs on traffic that matches the firewall rule. Valid values are allow or deny. Default is deny.
description	body	string	A human-readable description of the firewall rule.
destination_ip_addresses	body	string	The destination IPv4 or IPv6 address or CIDR for the firewall rule. No default.
destination_port	body	string	The destination port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
enabled	body	boolean	Set to false to disable this rule in the firewall policy. Facilitates selectively turning



Name	In	Type	Description
			off rules without having to disassociate the rule from the firewall policy. Valid values are true or false. Default is true.
firewall_policy_id	body	string	The ID of the firewall policy.
id	body	string	The ID of the firewall rule.
ip_version	body	integer	The IP protocol version for the firewall rule. Valid values are 4 or 6. Default is 4.
name	body	string	A human-readable name of the firewall rule.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol for the firewall rule. Possible values are icmp, tcp, udp, or null.
public	body	boolean	Indicates whether this firewall rule is shared across all projects.
source_ip_address	body	string	The source IPv4 or IPv6 address or CIDR for the firewall rule. No default.
source_port	body	string	The source port or port range for the firewall rule. A valid value is a port number, as an integer, or a port range, in the format of a : separated range. For a port range, include both ends of the range. For example, 80:90.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_ip_address": null,
    "destination_port": "80",
    "enabled": true,
    "firewall_policy_id": "c69933c1-b472-44f9-8226-30dc4ffd454c",
    "id": "8722e0e0-9cc9-4490-9660-8c9a5732fbb0",
    "ip_version": 4,
    "name": "ALLOW_HTTP",
    "position": 1,
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "protocol": "tcp",
    "public": true,
    "source_ip_address": null,
```

```
}
  "source_port": null,
  "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
}
```

## DELETE

/v2.0/fwaas/firewall\_rules/{firewall\_rule\_id}

### Delete firewall rule

Deletes a firewall rule. samples/firewall-v2/firewall-policy-create-response.json

Normal response codes: 204

Error response codes: 401, 404, 409

#### Request

Name	In	Type	Description
firewall_rule_id	path	string	The ID for the firewall rule.

## PUT

/v2.0/fwaas/firewall\_policies/{firewall\_policy\_id}/insert\_rule

### Insert rule into a firewall policy

Insert firewall rule into a policy.

A firewall\_rule\_id is inserted relative to the position of the firewall\_rule\_id set in insert\_before or insert\_after. If insert\_before is set, insert\_after is ignored. If both insert\_before and insert\_after are not set, the new firewall\_rule\_id is inserted as the first rule of the policy.

Normal response codes: 200

Error response codes: 400, 401, 404, 409

#### Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.
firewall_rule_id	body	string	The ID of the firewall rule.
insert_after	body	string	The ID of the firewall_rule to insert the new rule after. The new rule will be inserted immediately after the specified firewall_rule. If both before and after values are supplied, the after value will be ignored. To insert a rule into a policy with no rules yet, the both the before and the after values must be "".

Name	In	Type	Description
insert_before	body	string	The ID of the firewall_rule to insert the new rule before. The new rule will be inserted immediately before the specified firewall_rule. If both before and after values are supplied, the after value will be ignored. To insert a rule into a policy with no rules yet, the both the before and the after values must be "".

### Request Example

```
{
  "firewall_rule_id": "7bc34b8c-8d3b-4ada-a9c8-1f4c11c65692",
  "insert_after": "a08ef905-0ff6-4784-8374-175fffe7dade",
  "insert_before": ""
}
```

### Response Parameters

Name	In	Type	Description
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable name of the firewall policy.
firewall_rules	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
id	body	string	The ID of the firewall policy.
name	body	string	A human-readable name of the firewall policy.
project_id	body	string	The ID of the project.
public	body	boolean	Set to true to make this firewall policy visible to other projects. Default is false.
tenant_id	body	string	The ID of the project.

### Response Example

```
{
  "audited": false,
  "description": "",
  "firewall_rules": [
    "acbdfead-eca2-4456-838c-8b531e47b9c7"
  ],
}
```

```

    "id": "c9e15d6e-b6ba-4ef4-8715-985d1f100467",
    "name": "policy2",
    "public": false,
    "project_id": "95573613ec554b4b8df9f2679c64557b",
    "tenant_id": "95573613ec554b4b8df9f2679c64557b"
  }

```

**PUT**

/v2.0/fwaas/firewall\_policies/{firewall\_policy\_id}/remove\_rule

## Remove rule from firewall policy

Remove firewall rule from a policy.

Normal response codes: 200

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
firewall_policy_id	path	string	The ID of the firewall policy.
firewall_rule_id	body	string	The ID of the firewall rule.

### Request Example

```

{
  "firewall_rule_id": "7bc34b8c-8d3b-4ada-a9c8-1f4c11c65692"
}

```

### Response Parameters

Name	In	Type	Description
audited	body	boolean	Each time that the firewall policy or its associated rules are changed, the API sets this attribute to false. To audit the policy, explicitly set this attribute to true.
description	body	string	A human-readable name of the firewall policy.
firewall_rules	body	array	A list of the IDs of the firewall rules associated with the firewall policy.
id	body	string	The ID of the firewall policy.
name	body	string	A human-readable name of the firewall policy.
project_id	body	string	The ID of the project.
public	body	boolean	Set to true to make this firewall policy visible to other

Name	In	Type	Description
			projects. Default is false.
tenant_id	body	string	The ID of the project.

### Response Example

```
{
  "audited": false,
  "description": "",
  "firewall_rules": [],
  "id": "c9e15d6e-b6ba-4ef4-8715-985d1f100467",
  "name": "policy2",
  "project_id": "95573613ec554b4b8df9f2679c64557b",
  "public": false,
  "tenant_id": "95573613ec554b4b8df9f2679c64557b"
}
```

## Security group rules (security-group-rules)

Lists, creates, shows information for, and deletes security group rules.

**GET**

/v2.0/security-group-rules

### List security group rules

Lists a summary of all OpenStack Networking security group rules that the project can access.

The list provides the ID for each security group rule.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
security_group_rules	body	array	A list of security_group_rule objects. Refer to <a href="#">Security group rules</a> for details.
remote_group_id (Optional)	body	string	The remote group UUID to associate with this security group rule. You can specify either the remote_group_id or remote_ip_prefix attribute in the request body.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
ethertype (Optional)	body	string	Must be IPv4 or IPv6, and addresses represented in CIDR must match the ingress or egress rules.
port_range_max (Optional)	body	integer	The maximum port number in the range that is matched by the security group rule. The port_range_min attribute constrains the port_range_max attribute. If the protocol is ICMP, this value must be an ICMP type.
security_group_id	body	string	The security group ID to associate with this security group rule.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
port_range_min (Optional)	body	integer	The minimum port number in the range that is matched by the security group rule. If the protocol is TCP or UDP, this value must be less than or equal to the port_range_max attribute value. If the protocol is ICMP, this value must be an ICMP type.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
id	body	string	The ID of the security group rule.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "security_group_rules": [
    {
      "direction": "egress",
      "ethertype": "IPv6",
      "id": "3c0e45ff-adaf-4124-b083-bf390e5482ff",
      "port_range_max": null,
      "port_range_min": null,

```

```

        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    },
    {
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "93aa42e5-80db-4581-9391-3a608bd0e448",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    },
    {
        "direction": "ingress",
        "ethertype": "IPv6",
        "id": "c0b09f00-1d49-4e64-a0a7-8a186d928138",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    },
    {
        "direction": "ingress",
        "ethertype": "IPv4",
        "id": "f7d45c89-008e-4bab-88ad-d6811724c51c",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    }
]
}

```

**POST**

/v2.0/security-group-rules

## Create security group rule

Creates an OpenStack Networking security group rule.

Normal response codes: 201

Error response codes: 400, 401, 404, 409

## Request

Name	In	Type	Description
security_group_rule	body	object	A security_group_rule object.
remote_group_id (Optional)	body	string	The remote group UUID to associate with this security group rule. You can specify either the remote_group_id or remote_ip_prefix attribute in the request body.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
ethertype (Optional)	body	string	Must be IPv4 or IPv6, and addresses represented in CIDR must match the ingress or egress rules.
port_range_max (Optional)	body	integer	The maximum port number in the range that is matched by the security group rule. The port_range_min attribute constrains the port_range_max attribute. If the protocol is ICMP, this value must be an ICMP type.
security_group_id	body	string	The security group ID to associate with this security group rule.
port_range_min (Optional)	body	integer	The minimum port number in the range that is matched by the security group rule. If the protocol is TCP or UDP, this value must be less than or equal to the port_range_max attribute value. If the protocol is ICMP, this value must be an ICMP type.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{  
  "security_group_rule": {
```



```

"direction": "ingress",
"port_range_min": "80",
"ethertype": "IPv4",
"port_range_max": "80",
"protocol": "tcp",
"remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
"security_group_id": "a7734e61-b545-452d-a3cd-0189cbd9747a"
}

```

## Response Parameters

Name	In	Type	Description
security_group_rule	body	object	A security_group_rule object.
remote_group_id (Optional)	body	string	The remote group UUID to associate with this security group rule. You can specify either the remote_group_id or remote_ip_prefix attribute in the request body.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
ethertype (Optional)	body	string	Must be IPv4 or IPv6, and addresses represented in CIDR must match the ingress or egress rules.
port_range_max (Optional)	body	integer	The maximum port number in the range that is matched by the security group rule. The port_range_min attribute constrains the port_range_max attribute. If the protocol is ICMP, this value must be an ICMP type.
security_group_id	body	string	The security group ID to associate with this security group rule.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
port_range_min (Optional)	body	integer	The minimum port number in the range that is matched by the security group rule. If the protocol is TCP or UDP, this value must be less than or equal to the port_range_max attribute value. If the protocol is ICMP, this value must be an

Name	In	Type	Description
			ICMP type.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
id	body	string	The ID of the security group rule.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "security_group_rule": {
    "direction": "ingress",
    "ethertype": "IPv4",
    "id": "2bc0accf-312e-429a-956e-e4407625eb62",
    "port_range_max": 80,
    "port_range_min": 80,
    "protocol": "tcp",
    "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "remote_ip_prefix": null,
    "security_group_id": "a7734e61-b545-452d-a3cd-0189cbd9747a",
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "description": ""
  }
}
```

## GET

/v2.0/security-group-rules/{security\_group\_rule\_id}

## Show security group rule

Shows detailed information for a security group rule.

The response body contains the following information about the security group rule:

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
security_group_rule_id	path	string	The ID of the security group rule.
verbose (Optional)	query	boolean	Show detailed information.

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
<code>security_group_rule</code>	body	object	A <code>security_group_rule</code> object.
<code>remote_group_id</code> (Optional)	body	string	The remote group UUID to associate with this security group rule. You can specify either the <code>remote_group_id</code> or <code>remote_ip_prefix</code> attribute in the request body.
<code>direction</code>	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
<code>protocol</code> (Optional)	body	string	The IP protocol. Valid value is <code>icmp</code> , <code>tcp</code> , <code>udp</code> , or <code>null</code> . No default.
<code>ethertype</code> (Optional)	body	string	Must be IPv4 or IPv6, and addresses represented in CIDR must match the ingress or egress rules.
<code>port_range_max</code> (Optional)	body	integer	The maximum port number in the range that is matched by the security group rule. The <code>port_range_min</code> attribute constrains the <code>port_range_max</code> attribute. If the protocol is ICMP, this value must be an ICMP type.
<code>security_group_id</code>	body	string	The security group ID to associate with this security group rule.
<code>tenant_id</code>	body	string	The ID of the project.
<code>project_id</code>	body	string	The ID of the project.
<code>port_range_min</code> (Optional)	body	integer	The minimum port number in the range that is matched by the security group rule. If the protocol is TCP or UDP, this value must be less than or equal to the <code>port_range_max</code> attribute value. If the protocol is ICMP, this value must be an ICMP type.

Name	In	Type	Description
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
id	body	string	The ID of the security group rule.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "security_group_rule": {
    "direction": "egress",
    "ethertype": "IPv6",
    "id": "3c0e45ff-adaf-4124-b083-bf390e5482ff",
    "port_range_max": null,
    "port_range_min": null,
    "protocol": null,
    "remote_group_id": null,
    "remote_ip_prefix": null,
    "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550"
  }
}
```

## DELETE

/v2.0/security-group-rules/{security\_group\_rule\_id}

## Delete security group rule

Deletes a rule from an OpenStack Networking security group.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
security_group_rule_id	path	string	The ID of the security group rule.

## Response

There is no body content is returned on a successful DELETE request.

# Security groups (security-groups)

Lists, creates, shows information for, updates, and deletes security groups.

## GET

/v2.0/security-groups

## List security groups

Lists OpenStack Networking security groups to which the project has access.

The response is an array of `security_group` objects which contains a list of `security_group_rules` objects.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
security_groups	body	array	A list of <code>security_group</code> objects.
id	body	string	The ID of the security group.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
security_group_rules	body	array	A list of <code>security_group_rule</code> objects. Refer to <a href="#">Security group rules</a> for details.

### Response Example

```
{
  "security_groups": [
    {
      "description": "default",
      "id": "85cc3048-abc3-43cc-89b3-377341426ac5",
      "name": "default",
```

```

"security_group_rules": [
  {
    "direction": "egress",
    "ethertype": "IPv6",
    "id": "3c0e45ff-adaf-4124-b083-bf390e5482ff",
    "port_range_max": null,
    "port_range_min": null,
    "protocol": null,
    "remote_group_id": null,
    "remote_ip_prefix": null,
    "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "description": ""
  },
  {
    "direction": "egress",
    "ethertype": "IPv4",
    "id": "93aa42e5-80db-4581-9391-3a608bd0e448",
    "port_range_max": null,
    "port_range_min": null,
    "protocol": null,
    "remote_group_id": null,
    "remote_ip_prefix": null,
    "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "description": ""
  },
  {
    "direction": "ingress",
    "ethertype": "IPv6",
    "id": "c0b09f00-1d49-4e64-a0a7-8a186d928138",
    "port_range_max": null,
    "port_range_min": null,
    "protocol": null,
    "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "remote_ip_prefix": null,
    "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "description": ""
  },
  {
    "direction": "ingress",
    "ethertype": "IPv4",
    "id": "f7d45c89-008e-4bab-88ad-d6811724c51c",
    "port_range_max": null,
    "port_range_min": null,
    "protocol": null,
    "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "remote_ip_prefix": null,
    "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "description": ""
  }
],
"project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
"tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550"
}
]
}

```

**POST**

/v2.0/security-groups

## Create security group

Creates an OpenStack Networking security group.

This operation creates a security group with default security group rules for the IPv4 and IPv6 ether types.

Normal response codes: 201

Error response codes: 400, 401, 409

### Request

Name	In	Type	Description
security_group	body	object	A security_group object.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
name	body	string	Human-readable name of the resource.

### Request Example

```
{
  "security_group": {
    "name": "new-webservers",
    "description": "security group for webservers"
  }
}
```

### Response Parameters

Name	In	Type	Description
security_group	body	object	A security_group object.
id	body	string	The ID of the security group.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.

Name	In	Type	Description
security_group_rules	body	array	A list of security_group_rule objects. Refer to <a href="#">Security group rules</a> for details.

## Response Example

```
{
  "security_group": {
    "description": "security group for webserver",
    "id": "2076db17-a522-4506-91de-c6dd8e837028",
    "name": "new-webserver",
    "security_group_rules": [
      {
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "38ce2d8e-e8f1-48bd-83c2-d33cb9f50c3d",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "security_group_id": "2076db17-a522-4506-91de-c6dd8e837028",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
      },
      {
        "direction": "egress",
        "ethertype": "IPv6",
        "id": "565b9502-12de-4ffd-91e9-68885cfff6ae1",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "security_group_id": "2076db17-a522-4506-91de-c6dd8e837028",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
      }
    ],
    "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
    "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550"
  }
}
```

**GET**

/v2.0/security-groups/{security\_group\_id}

## Show security group

Shows details for a security group.

The associated security group rules are contained in the response.

Normal response codes: 200

Error response codes: 401, 404



## Request

Name	In	Type	Description
security_group_id	path	string	The ID of the security group.
verbose (Optional)	query	boolean	Show detailed information.
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Request Example

```
GET /v2.0/security-groups/85cc3048-abc3-43cc-89b3-377341426ac5
Accept: application/json
```

## Response Parameters

Name	In	Type	Description
security_group	body	object	A <code>security_group</code> object.
id	body	string	The ID of the security group.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
security_group_rules	body	array	A list of <code>security_group_rule</code> objects. Refer to <a href="#">Security group rules</a> for details.

## Response Example

```
{
  "security_group": {
    "description": "default",
    "id": "85cc3048-abc3-43cc-89b3-377341426ac5",
    "name": "default",
    "security_group_rules": [
      {
        "direction": "egress",
        "ethertype": "IPv6",
        "id": "3c0e45ff-adaf-4124-b083-bf390e5482ff",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,

```

```

        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    },
    {
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "93aa42e5-80db-4581-9391-3a608bd0e448",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    },
    {
        "direction": "ingress",
        "ethertype": "IPv6",
        "id": "c0b09f00-1d49-4e64-a0a7-8a186d928138",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    },
    {
        "direction": "ingress",
        "ethertype": "IPv4",
        "id": "f7d45c89-008e-4bab-88ad-d6811724c51c",
        "port_range_max": null,
        "port_range_min": null,
        "protocol": null,
        "remote_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "remote_ip_prefix": null,
        "security_group_id": "85cc3048-abc3-43cc-89b3-377341426ac5",
        "project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550",
        "description": ""
    }
],
"project_id": "e4f50856753b4dc6afee5fa6b9b6c550",
"tenant_id": "e4f50856753b4dc6afee5fa6b9b6c550"
}
}

```

**PUT**

/v2.0/security-groups/{security\_group\_id}

## Update security group

Updates a security group.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
security_group_id	path	string	The ID of the security group.
security_group	body	object	A security_group object.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
name	body	string	Human-readable name of the resource.

## Request Example

```
{
  "security_group": {
    "name": "mysecgroup",
    "description": "my security group"
  }
}
```

## Response Parameters

Name	In	Type	Description
security_group	body	object	A security_group object.
id	body	string	The ID of the security group.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
security_group_rules	body	array	A list of security_group_rule objects. Refer to <a href="#">Security group rules</a> for details.

## Response Example

```
{
  "security_group": {
    "security_group_rules": [],
    "project_id": "a52cdb9cc7854a39a23d3af73a40899e",
    "tenant_id": "a52cdb9cc7854a39a23d3af73a40899e",
    "id": "01fbade5-b664-42f6-83ae-4e214f4263fa",
    "name": "mysecgroup",
    "description": "my security group"
  }
}
```

## DELETE

/v2.0/security-groups/{security\_group\_id}

## Delete security group

Deletes an OpenStack Networking security group.

This operation deletes an OpenStack Networking security group and its associated security group rules, provided that a port is not associated with the security group. If a port is associated with the security group 409 (Conflict) is returned.

This operation does not require a request body. This operation does not return a response body.

Normal response codes: 204

Error response codes: 401, 404, 409

#### Request

Name	In	Type	Description
security_group_id	path	string	The ID of the security group.

#### Request Example

```
DELETE /v2.0/security-groups/e470bdfc-4869-459b-a561-cb3377efae59
Content-Type: application/json
Accept: application/json
```

#### Response

There is no body content for the response of a successful DELETE request.

## VPNaaS 2.0 (UNMAINTAINED) (vpn, vpnservices, ikepolicies, ipsecpolicies, endpoint-groups, ipsec-site-connections)

The Virtual-Private-Network-as-a-Service (VPNaaS) extension enables OpenStack projects to extend private networks across the public telecommunication infrastructure.

This initial implementation of the VPNaaS extension provides:

- Site-to-site VPN that connects two private networks.
- Multiple VPN connections per project.
- IKEv1 policy support with 3des, aes-128, aes-256, or aes-192 encryption.
- IPSec policy support with 3des, aes-128, aes-192, or aes-256 encryption, sha1 authentication, ESP, AH, or AH-ESP transform protocol, and tunnel or transport mode encapsulation.
- Dead Peer Detection (DPD) with hold, clear, restart, disabled, or restart-by-peer actions.

This extension introduces these resources:

- service. A parent object that associates VPN with a specific subnet and router.
- ikepolicy. The Internet Key Exchange (IKE) policy that identifies the

authentication and encryption algorithm to use during phase one and two negotiation of a VPN connection.

- `ipsecpolicy`. The IP security policy that specifies the authentication and encryption algorithm and encapsulation mode to use for the established VPN connection.
- `ipsec-site-connection`. Details for the site-to-site IPsec connection, including the peer CIDRs, MTU, authentication mode, peer address, DPD settings, and status.
- `endpoint-groups`. Defines one or more endpoints of a specific type, and can be used to specify both local and peer endpoints for IPsec Connections.

**GET**

`/v2.0/vpn/ikepolicies`

## List IKE policies

Lists IKE policies.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
ikepolicies	body	array	A list of <code>ikepolicy</code> objects.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description

Name	In	Type	Description
			for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
phase1_negotiation_mode (Optional)	body	string	The IKE mode. A valid value is main, which is the default.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IKE policy.
ike_version (Optional)	body	string	The IKE version. A valid value is v1 or v2. Default is v1.

## Response Example

```
{
  "ikepolicies": [
```

```

{
  "name": "ikepolicy1",
  "project_id": "ccb81365fe36411a9011e90491fe1330",
  "tenant_id": "ccb81365fe36411a9011e90491fe1330",
  "auth_algorithm": "sha1",
  "encryption_algorithm": "aes-256",
  "pfs": "group5",
  "phase1_negotiation_mode": "main",
  "lifetime": {
    "units": "seconds",
    "value": 3600
  },
  "ike_version": "v1",
  "id": "5522aff7-1b3c-48dd-9c3c-b50f016b73db",
  "description": ""
}
]
}

```

**POST**

/v2.0/vpn/ikepolicies

## Create IKE policy

Creates an IKE policy.

The IKE policy is used for phases one and two negotiation of the VPN connection. You can specify both the authentication and encryption algorithms for connections.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
ikepolicy	body	object	An ikepolicy object.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A

Name	In	Type	Description
			valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
phase1_negotiation_mode (Optional)	body	string	The IKE mode. A valid value is main, which is the default.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
ike_version (Optional)	body	string	The IKE version. A valid value is v1 or v2. Default is v1.

### Request Example

```
{
  "ikepolicy": {
    "phase1_negotiation_mode": "main",
    "auth_algorithm": "sha1",
    "encryption_algorithm": "aes-128",
    "pfs": "group5",
    "lifetime": {
      "units": "seconds",
      "value": 7200
    },
    "ike_version": "v1",
    "name": "ikepolicy1"
  }
}
```

### Response Parameters

Name	In	Type	Description
ikepolicies	body	array	A list of ikepolicy objects.
ikepolicy	body	object	An ikepolicy object.



Name	In	Type	Description
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
phase1_negotiation_mode (Optional)	body	string	The IKE mode. A valid value is main, which is the default.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IKE policy.

Name	In	Type	Description
ike_version (Optional)	body	string	The IKE version. A valid value is v1 or v2. Default is v1.

## Response Example

```
{
  "ikepolicy": {
    "name": "ikepolicy1",
    "project_id": "ccb81365fe36411a9011e90491fe1330",
    "tenant_id": "ccb81365fe36411a9011e90491fe1330",
    "auth_algorithm": "sha1",
    "encryption_algorithm": "aes-128",
    "pfs": "group5",
    "phase1_negotiation_mode": "main",
    "lifetime": {
      "units": "seconds",
      "value": 7200
    },
    "ike_version": "v1",
    "id": "5522aff7-1b3c-48dd-9c3c-b50f016b73db",
    "description": ""
  }
}
```

## GET

/v2.0/vpn/ikepolicies/{ikepolicy\_id}

## Show IKE policy details

Shows details for an IKE policy.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
ikepolicy_id	path	string	The ID of the IKE policy.

## Response Parameters

Name	In	Type	Description
ikepolicies	body	array	A list of ikepolicy objects.
ikepolicy	body	object	An ikepolicy object.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.

Name	In	Type	Description
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
phase1_negotiation_mode (Optional)	body	string	The IKE mode. A valid value is main, which is the default.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IKE policy.
ike_version (Optional)	body	string	The IKE version. A valid value is v1 or v2. Default is v1.

## Response Example

```
{
  "ikepolicy": {
    "name": "ikepolicy1",
    "project_id": "ccb81365fe36411a9011e90491fe1330",
    "tenant_id": "ccb81365fe36411a9011e90491fe1330",
    "auth_algorithm": "sha1",
    "encryption_algorithm": "aes-256",
    "pfs": "group5",
    "phase1_negotiation_mode": "main",
    "lifetime": {
      "units": "seconds",
      "value": 3600
    },
    "ike_version": "v1",
    "id": "5522aff7-1b3c-48dd-9c3c-b50f016b73db",
    "description": ""
  }
}
```

}

**PUT**

/v2.0/vpn/ikepolicies/{ikepolicy\_id}

## Update IKE policy

Updates policy settings in an IKE policy.

Normal response codes: 200

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
ikepolicy_id	path	string	The ID of the IKE policy.
ikepolicy	body	object	An ikepolicy object.
description	body	string	A human-readable description for the resource.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
name	body	string	Human-readable name of the resource.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
phase1_negotiation_mode (Optional)	body	string	The IKE mode. A valid value is main, which is the default.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit

Name	In	Type	Description
			is seconds and default value is 3600.
ike_version (Optional)	body	string	The IKE version. A valid value is v1 or v2. Default is v1.

### Request Example

```
{
  "ikepolicy": {
    "encryption_algorithm": "aes-256"
  }
}
```

### Response Parameters

Name	In	Type	Description
ikepolicies	body	array	A list of ikempolicy objects.
ikempolicy	body	object	An ikempolicy object.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
name	body	string	Human-readable name of the resource.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
phase1_negotiation_mode (Optional)	body	string	The IKE mode. A valid value is main, which is the default.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value

Name	In	Type	Description
			is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IKE policy.
ike_version (Optional)	body	string	The IKE version. A valid value is v1 or v2. Default is v1.

## Response Example

```
{
  "ikepolicy": {
    "name": "ikepolicy1",
    "project_id": "ccb81365fe36411a9011e90491fe1330",
    "tenant_id": "ccb81365fe36411a9011e90491fe1330",
    "auth_algorithm": "sha1",
    "encryption_algorithm": "aes-256",
    "pfs": "group5",
    "phase1_negotiation_mode": "main",
    "lifetime": {
      "units": "seconds",
      "value": 3600
    },
    "ike_version": "v1",
    "id": "5522aff7-1b3c-48dd-9c3c-b50f016b73db",
    "description": ""
  }
}
```

## DELETE

/v2.0/vpn/ikepolicies/{ikepolicy\_id}

## Remove IKE policy

Removes an IKE policy.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
ikepolicy_id	path	string	The ID of the IKE policy.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/vpn/ipsecpolicies

## List IPsec policies

Lists all IPsec policies.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
ipsecpolicies	body	array	A list of ipsecpolicy objects.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encapsulation_mode (Optional)	body	string	The encapsulation mode. A valid value is tunnel or transport. Default is tunnel.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer

Name	In	Type	Description
			value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
transform_protocol (Optional)	body	string	The transform protocol. A valid value is ESP, AH, or AH- ESP. Default is ESP.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IPSec policy.
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "ipsecpolicies": [
    {
      "name": "ipsecpolicy1",
      "transform_protocol": "esp",
      "auth_algorithm": "sha1",
      "encapsulation_mode": "tunnel",
      "encryption_algorithm": "aes-128",
      "pfs": "group14",
      "project_id": "ccb81365fe36411a9011e90491fe1330",
      "tenant_id": "ccb81365fe36411a9011e90491fe1330",
      "lifetime": {
        "units": "seconds",
        "value": 3600
      },
      "id": "5291b189-fd84-46e5-84bd-78f40c05d69c",
      "description": ""
    }
  ]
}
```

**POST**

/v2.0/vpn/ipsecpolicies

## Create IPSec policy

Creates an IP security (IPSec) policy.

The IPsec policy specifies the authentication and encryption algorithms and encapsulation mode to use for the established VPN connection.

Normal response codes: 201



## Error response codes: 400, 401

### Request

Name	In	Type	Description
ipsecpolicy	body	object	An ipsecpolicy object.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encapsulation_mode (Optional)	body	string	The encapsulation mode. A valid value is tunnel or transport. Default is tunnel.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
transform_protocol (Optional)	body	string	The transform protocol. A valid value is ESP, AH, or AH- ESP. Default is ESP.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
name	body	string	Human-readable name of the resource.

### Request Example

```
{  
  "ipsecpolicy": {  
    "name": "ipsecpolicy1",
```

```

    "transform_protocol": "esp",
    "auth_algorithm": "sha1",
    "encapsulation_mode": "tunnel",
    "encryption_algorithm": "aes-128",
    "pfs": "group5",
    "lifetime": {
        "units": "seconds",
        "value": 7200
    }
}
}
}

```

## Response Parameters

Name	In	Type	Description
ipsecpolicies	body	array	A list of ipsecpolicy objects.
ipsecpolicy	body	object	An ipsecpolicy object.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encapsulation_mode (Optional)	body	string	The encapsulation mode. A valid value is tunnel or transport. Default is tunnel.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
transform_protocol (Optional)	body	string	The transform protocol. A valid value is ESP, AH, or AH- ESP. Default is ESP.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.

Name	In	Type	Description
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IPSec policy.
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "ipsecpolicy": {
    "name": "ipsecpolicy1",
    "transform_protocol": "esp",
    "auth_algorithm": "sha1",
    "encapsulation_mode": "tunnel",
    "encryption_algorithm": "aes-128",
    "pfs": "group5",
    "project_id": "ccb81365fe36411a9011e90491fe1330",
    "tenant_id": "ccb81365fe36411a9011e90491fe1330",
    "lifetime": {
      "units": "seconds",
      "value": 7200
    },
    "id": "5291b189-fd84-46e5-84bd-78f40c05d69c",
    "description": ""
  }
}
```

**GET**

/v2.0/vpn/ipsecpolicies/{ipsecpolicy\_id}

## Show IPSec policy

Shows details for an IPSec policy.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
ipsecpolicy_id (Optional)	path	string	The ID of the IPSec policy.

## Response Parameters

Name	In	Type	Description
ipsecpolicies	body	array	A list of ipsecpolicy objects.
ipsecpolicy	body	object	An ipsecpolicy object.
description	body	string	A human-readable description

Name	In	Type	Description
			for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encapsulation_mode (Optional)	body	string	The encapsulation mode. A valid value is tunnel or transport. Default is tunnel.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
transform_protocol (Optional)	body	string	The transform protocol. A valid value is ESP, AH, or AH-ESP. Default is ESP.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IPsec policy.

Name	In	Type	Description
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "ipsecpolicy": {
    "name": "ipsecpolicy1",
    "transform_protocol": "esp",
    "auth_algorithm": "sha1",
    "encapsulation_mode": "tunnel",
    "encryption_algorithm": "aes-128",
    "pfs": "group14",
    "project_id": "ccb81365fe36411a9011e90491fe1330",
    "tenant_id": "ccb81365fe36411a9011e90491fe1330",
    "lifetime": {
      "units": "seconds",
      "value": 3600
    },
    "id": "5291b189-fd84-46e5-84bd-78f40c05d69c",
    "description": ""
  }
}
```

## PUT

/v2.0/vpn/ipsecpolicies/{ipsecpolicy\_id}

## Update IPSec policy

Updates policy settings in an IPSec policy.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
ipsecpolicy_id (Optional)	path	string	The ID of the IPSec policy.
ipsecpolicy	body	object	An ipsecpolicy object.
description	body	string	A human-readable description for the resource.
transform_protocol (Optional)	body	string	The transform protocol. A valid value is ESP, AH, or AH-ESP. Default is ESP.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encapsulation_mode (Optional)	body	string	The encapsulation mode. A valid value is tunnel or

Name	In	Type	Description
			transport. Default is tunnel.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
name	body	string	Human-readable name of the resource.

### Request Example

```
{
  "ipsecpolicy": {
    "pfs": "group14"
  }
}
```

### Response Parameters

Name	In	Type	Description
ipsecpolicies	body	array	A list of ipsecpolicy objects.
ipsecpolicy	body	object	An ipsecpolicy object.

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
auth_algorithm (Optional)	body	string	The authentication hash algorithm. Valid values are sha1, sha256, sha384, sha512. The default is sha1.
encapsulation_mode (Optional)	body	string	The encapsulation mode. A valid value is tunnel or transport. Default is tunnel.
encryption_algorithm (Optional)	body	string	The encryption algorithm. A valid value is 3des, aes-128, aes-192, aes-256, and so on. Default is aes-128.
pfs (Optional)	body	string	Perfect forward secrecy (PFS). A valid value is Group2, Group5, Group14, and so on. Default is Group5.
value (Optional)	body	integer	The lifetime value, as a positive integer. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
transform_protocol (Optional)	body	string	The transform protocol. A valid value is ESP, AH, or AH-ESP. Default is ESP.
units (Optional)	body	string	The units for the lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
lifetime (Optional)	body	object	The lifetime of the security association. The lifetime consists of a unit and integer value. You can omit either the unit or value portion of the lifetime. Default unit is seconds and default value is 3600.
id	body	string	The ID of the IPSec policy.

Name	In	Type	Description
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "ipsecpolicy": {
    "name": "ipsecpolicy1",
    "transform_protocol": "esp",
    "auth_algorithm": "sha1",
    "encapsulation_mode": "tunnel",
    "encryption_algorithm": "aes-128",
    "pfs": "group14",
    "project_id": "ccb81365fe36411a9011e90491fe1330",
    "tenant_id": "ccb81365fe36411a9011e90491fe1330",
    "lifetime": {
      "units": "seconds",
      "value": 3600
    },
    "id": "5291b189-fd84-46e5-84bd-78f40c05d69c",
    "description": ""
  }
}
```

## DELETE

/v2.0/vpn/ipsecpolicies/{ipsecpolicy\_id}

## Remove IPSec policy

Removes an IPSec policy.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
ipsecpolicy_id (Optional)	path	string	The ID of the IPSec policy.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/vpn/ipsec-site-connections

## List IPSec connections

Lists all IPSec connections.

Use the `fields` query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).



Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using fields parameter, the API returns only the requested set of attributes. fields parameter can be specified multiple times. For example, if you specify fields=id&fields=name in the request URL, only id and name attributes will be returned.

## Response Parameters

Name	In	Type	Description
auth_mode (Optional)	body	string	The authentication mode. A valid value is psk, which is the default.
ikepolicy_id	body	string	The ID of the IKE policy.
vpnservice_id	body	string	The ID of the VPN service.
local_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private subnets for the local side of the connection. You must specify this parameter with the peer_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
peer_address	body	string	The peer gateway public IPv4 or IPv6 address or FQDN.
id (Optional)	body	string	The ID of the IPSec site-to-site connection.
route_mode (Optional)	body	string	The route mode. A valid value is static, which is the default.
ipsecpolicy_id	body	string	The ID of the IPSec policy.
peer_id	body	string	The peer router identity for authentication. A valid value is an IPv4 address, IPv6 address, e-mail address, key ID, or FQDN. Typically, this value matches the peer_address value.
status	body	string	Indicates whether the IPSec connection is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.

Name	In	Type	Description
psk	body	string	The pre-shared key. A valid value is any string.
description	body	string	A human-readable description for the resource.
initiator (Optional)	body	string	Indicates whether this VPN can only respond to connections or both respond to and initiate connections. A valid value is response- only or bi-directional. Default is bi-directional.
peer_cidrs (Optional)	body	array	(Deprecated) Unique list of valid peer private CIDRs in the form < net_address > / < prefix > .
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
interval (Optional)	body	integer	The dead peer detection (DPD) interval, in seconds. A valid value is a positive integer. Default is 30.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
peer_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private CIDRs in the form < net_address > / < prefix > for the peer side of the connection. You must specify this parameter with the local_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
dpd (Optional)	body	object	A dictionary with dead peer detection (DPD) protocol controls.
timeout	body	integer	The dead peer detection (DPD) timeout in seconds. A valid value is a positive integer that is greater than the DPD interval value. Default is 120.
action	body	string	The dead peer detection (DPD) action. A valid value is clear, hold, restart, disabled, or restart-by-peer. Default value is hold.
local_id (Optional)	body	string	An ID to be used instead of the external IP address for a virtual router used in traffic between instances on different networks in east-west traffic. Most often, local ID would be domain name, email address, etc. If this is not configured then the external IP address will be used as the ID.

## Response Example

```
{
```

```

"ipsec_site_connections": [
  {
    "status": "PENDING CREATE",
    "psk": "secret",
    "initiator": "bi-directional",
    "name": "vpnconnection1",
    "admin_state_up": true,
    "project_id": "10039663455a446d8ba2cbb058b0f578",
    "tenant_id": "10039663455a446d8ba2cbb058b0f578",
    "auth_mode": "psk",
    "peer_cidrs": [],
    "mtu": 1500,
    "peer_ep_group_id": "9ad5a7e0-6dac-41b4-b20d-a7b8645fddf1",
    "ikepolicy_id": "9b00d6b0-6c93-4ca5-9747-b8ade7bb514f",
    "vpnservice_id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "dpd": {
      "action": "hold",
      "interval": 30,
      "timeout": 120
    },
    "route_mode": "static",
    "ipsecpolicy_id": "e6e23d0c-9519-4d52-8ea4-5b1f96d857b1",
    "local_ep_group_id": "3e1815dd-e212-43d0-8f13-b494fa553e68",
    "peer_address": "172.24.4.226",
    "peer_id": "172.24.4.226",
    "id": "851f280f-5639-4ea3-81aa-e298525ab74b",
    "description": ""
  }
]
}

```

**POST**

/v2.0/vpn/ipsec-site-connections

## Create IPSec connection

Creates a site-to-site IPSec connection for a service.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
ipsec_site_connection	body	object	An ipsec_site_connection object.
auth_mode (Optional)	body	string	The authentication mode. A valid value is psk, which is the default.
ikepolicy_id (Optional)	body	string	The ID of the IKE policy.
vpnservice_id (Optional)	body	string	The ID of the VPN service.
local_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private subnets for the local side of the connection. You must specify this parameter with the peer_ep_group_id parameter unless

Name	In	Type	Description
			in backward- compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
peer_address	body	string	The peer gateway public IPv4 or IPv6 address or FQDN.
route_mode (Optional)	body	string	The route mode. A valid value is static, which is the default.
ipsecpolicy_id (Optional)	body	string	The ID of the IPsec policy.
peer_id	body	string	The peer router identity for authentication. A valid value is an IPv4 address, IPv6 address, e-mail address, key ID, or FQDN. Typically, this value matches the peer_address value.
psk	body	string	The pre-shared key. A valid value is any string.
description	body	string	A human-readable description for the resource.
initiator (Optional)	body	string	Indicates whether this VPN can only respond to connections or both respond to and initiate connections. A valid value is response- only or bi-directional. Default is bi-directional.
peer_cidrs (Optional)	body	array	(Deprecated) Unique list of valid peer private CIDRs in the form < net_address > / < prefix > .
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
interval (Optional)	body	integer	The dead peer detection (DPD) interval, in seconds. A valid value is a positive integer. Default is 30.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
peer_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private CIDRs in the form < net_address > / < prefix > for the peer side of the connection. You must specify this parameter with the local_ep_group_id parameter unless in backward-compatible mode where

Name	In	Type	Description
			peer_cidrs is provided with a subnet_id for the VPN service.
dpd (Optional)	body	object	A dictionary with dead peer detection (DPD) protocol controls.
timeout	body	integer	The dead peer detection (DPD) timeout in seconds. A valid value is a positive integer that is greater than the DPD interval value. Default is 120.
action	body	string	The dead peer detection (DPD) action. A valid value is clear, hold, restart, disabled, or restart-by-peer. Default value is hold.
local_id (Optional)	body	string	An ID to be used instead of the external IP address for a virtual router used in traffic between instances on different networks in east-west traffic. Most often, local ID would be domain name, email address, etc. If this is not configured then the external IP address will be used as the ID.

### Request Example

```
{
  "ipsec_site_connection": {
    "psk": "secret",
    "initiator": "bi-directional",
    "ipsecpolicy_id": "e6e23d0c-9519-4d52-8ea4-5b1f96d857b1",
    "admin_state_up": true,
    "mtu": "1500",
    "peer_ep_group_id": "9ad5a7e0-6dac-41b4-b20d-a7b8645fddf1",
    "ikepolicy_id": "9b00d6b0-6c93-4ca5-9747-b8ade7bb514f",
    "vpnservice_id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "local_ep_group_id": "3e1815dd-e212-43d0-8f13-b494fa553e68",
    "peer_address": "172.24.4.233",
    "peer_id": "172.24.4.233",
    "name": "vpnconnection1"
  }
}
```

### Response Parameters

Name	In	Type	Description
ipsec_site_connection	body	object	An ipsec_site_connection object.
auth_mode (Optional)	body	string	The authentication mode. A valid value is psk, which is the default.
ikepolicy_id	body	string	The ID of the IKE policy.
vpnservice_id	body	string	The ID of the VPN service.
local_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private subnets for the local side of the connection. You must specify this parameter with

Name	In	Type	Description
			the peer_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
peer_address	body	string	The peer gateway public IPv4 or IPv6 address or FQDN.
id (Optional)	body	string	The ID of the IPsec site-to-site connection.
route_mode (Optional)	body	string	The route mode. A valid value is static, which is the default.
ipsecpolicy_id	body	string	The ID of the IPsec policy.
peer_id	body	string	The peer router identity for authentication. A valid value is an IPv4 address, IPv6 address, e-mail address, key ID, or FQDN. Typically, this value matches the peer_address value.
status	body	string	Indicates whether the IPsec connection is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
psk	body	string	The pre-shared key. A valid value is any string.
description	body	string	A human-readable description for the resource.
initiator (Optional)	body	string	Indicates whether this VPN can only respond to connections or both respond to and initiate connections. A valid value is response-only or bi-directional. Default is bi-directional.
peer_cidrs (Optional)	body	array	(Deprecated) Unique list of valid peer private CIDRs in the form < net_address > / < prefix > .
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
interval (Optional)	body	integer	The dead peer detection (DPD) interval, in seconds. A valid value is a positive integer. Default is 30.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
peer_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private CIDRs in the form < net_address > / < prefix > for the peer side of the connection. You

Name	In	Type	Description
			must specify this parameter with the local_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
dpd (Optional)	body	object	A dictionary with dead peer detection (DPD) protocol controls.
timeout	body	integer	The dead peer detection (DPD) timeout in seconds. A valid value is a positive integer that is greater than the DPD interval value. Default is 120.
action	body	string	The dead peer detection (DPD) action. A valid value is clear, hold, restart, disabled, or restart-by-peer. Default value is hold.
local_id (Optional)	body	string	An ID to be used instead of the external IP address for a virtual router used in traffic between instances on different networks in east-west traffic. Most often, local ID would be domain name, email address, etc. If this is not configured then the external IP address will be used as the ID.

## Response Example

```
{
  "ipsec_site_connection": {
    "status": "PENDING_CREATE",
    "psk": "secret",
    "initiator": "bi-directional",
    "name": "vpnconnection1",
    "admin_state_up": true,
    "project_id": "10039663455a446d8ba2cbb058b0f578",
    "tenant_id": "10039663455a446d8ba2cbb058b0f578",
    "auth_mode": "psk",
    "peer_cidrs": [],
    "mtu": 1500,
    "peer_ep_group_id": "9ad5a7e0-6dac-41b4-b20d-a7b8645fddf1",
    "ikepolicy_id": "9b00d6b0-6c93-4ca5-9747-b8ade7bb514f",
    "vpnservice_id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "dpd": {
      "action": "hold",
      "interval": 30,
      "timeout": 120
    },
    "route_mode": "static",
    "ipsecpolicy_id": "e6e23d0c-9519-4d52-8ea4-5b1f96d857b1",
    "local_ep_group_id": "3e1815dd-e212-43d0-8f13-b494fa553e68",
    "peer_address": "172.24.4.233",
    "peer_id": "172.24.4.233",
    "id": "851f280f-5639-4ea3-81aa-e298525ab74b",
    "description": ""
  }
}
```

**GET**

/v2.0/vpn/ipsec-site-connections/{connection\_id}

## Show IPsec connection

Shows details for an IPsec connection.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
connection_id (Optional)	path	string	The ID of the IPsec site-to-site connection.

## Response Parameters

Name	In	Type	Description
auth_mode (Optional)	body	string	The authentication mode. A valid value is psk, which is the default.
ikepolicy_id	body	string	The ID of the IKE policy.
vpnservice_id	body	string	The ID of the VPN service.
local_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private subnets for the local side of the connection. You must specify this parameter with the peer_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
peer_address	body	string	The peer gateway public IPv4 or IPv6 address or FQDN.
id (Optional)	body	string	The ID of the IPsec site-to-site connection.
ipsec_site_connection	body	object	An ipsec_site_connection object.
route_mode (Optional)	body	string	The route mode. A valid value is static, which is the default.
ipsecpolicy_id	body	string	The ID of the IPsec policy.
peer_id	body	string	The peer router identity for authentication. A valid value is an IPv4 address, IPv6 address, e-mail address, key ID, or FQDN. Typically, this value matches the peer_address value.
status	body	string	Indicates whether the IPsec connection is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
psk	body	string	The pre-shared key. A valid value is any string.
description	body	string	A human-readable description for the resource.



Name	In	Type	Description
initiator (Optional)	body	string	Indicates whether this VPN can only respond to connections or both respond to and initiate connections. A valid value is response- only or bi-directional. Default is bi-directional.
peer_cidrs (Optional)	body	array	(Deprecated) Unique list of valid peer private CIDRs in the form < net_address > / < prefix > .
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
interval (Optional)	body	integer	The dead peer detection (DPD) interval, in seconds. A valid value is a positive integer. Default is 30.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
peer_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private CIDRs in the form < net_address > / < prefix > for the peer side of the connection. You must specify this parameter with the local_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
dpd (Optional)	body	object	A dictionary with dead peer detection (DPD) protocol controls.
timeout	body	integer	The dead peer detection (DPD) timeout in seconds. A valid value is a positive integer that is greater than the DPD interval value. Default is 120.
action	body	string	The dead peer detection (DPD) action. A valid value is clear, hold, restart, disabled, or restart-by-peer. Default value is hold.
local_id (Optional)	body	string	An ID to be used instead of the external IP address for a virtual router used in traffic between instances on different networks in east-west traffic. Most often, local ID would be domain name, email address, etc. If this is not configured then the external IP address will be used as the ID.

## Response Example

```
{
  "ipsec_site_connection": {
    "status": "DOWN",
    "psk": "secret",
    "initiator": "bi-directional",
    "name": "vpnconnection1",
    "admin_state_up": true,
    "project_id": "10039663455a446d8ba2cbb058b0f578",
    "tenant_id": "10039663455a446d8ba2cbb058b0f578",
    "auth_mode": "psk",
    "peer_cidrs": [],
    "mtu": 1500,
    "peer_ep_group_id": "9ad5a7e0-6dac-41b4-b20d-a7b8645fddf1",
    "ikepolicy_id": "9b00d6b0-6c93-4ca5-9747-b8ade7bb514f",
    "vpnservice_id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "dpd": {
      "action": "hold",
      "interval": 30,
      "timeout": 120
    },
    "route_mode": "static",
    "ipsecpolicy_id": "e6e23d0c-9519-4d52-8ea4-5b1f96d857b1",
    "local_ep_group_id": "3e1815dd-e212-43d0-8f13-b494fa553e68",
    "peer_address": "172.24.4.226",
    "peer_id": "172.24.4.226",
    "id": "851f280f-5639-4ea3-81aa-e298525ab74b",
    "description": ""
  }
}
```

**PUT**

/v2.0/vpn/ipsec-site-connections/{connection\_id}

## Update IPsec connection

Updates connection settings for an IPsec connection.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
connection_id (Optional)	path	string	The ID of the IPsec site-to-site connection.
ipsec_site_connection	body	object	An ipsec_site_connection object.
psk	body	string	The pre-shared key. A valid value is any string.
initiator (Optional)	body	string	Indicates whether this VPN can only respond to connections or both respond to and initiate connections. A valid value is response- only or bi-directional. Default is bi-directional.

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
interval (Optional)	body	integer	The dead peer detection (DPD) interval, in seconds. A valid value is a positive integer. Default is 30.
peer_cidrs (Optional)	body	array	(Deprecated) Unique list of valid peer private CIDRs in the form < net_address > / < prefix > .
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
peer_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private CIDRs in the form < net_address > / < prefix > for the peer side of the connection. You must specify this parameter with the local_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
local_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private subnets for the local side of the connection. You must specify this parameter with the peer_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
dpd (Optional)	body	object	A dictionary with dead peer detection (DPD) protocol controls.
timeout	body	integer	The dead peer detection (DPD) timeout in seconds. A valid value is a positive integer that is greater than the DPD interval value. Default is 120.
action	body	string	The dead peer detection (DPD) action. A valid value is clear, hold, restart, disabled, or restart-by-peer. Default value is hold.
peer_address	body	string	The peer gateway public IPv4 or IPv6 address or FQDN.

Name	In	Type	Description
peer_id	body	string	The peer router identity for authentication. A valid value is an IPv4 address, IPv6 address, e-mail address, key ID, or FQDN. Typically, this value matches the peer_address value.
name	body	string	Human-readable name of the resource.
local_id (Optional)	body	string	An ID to be used instead of the external IP address for a virtual router used in traffic between instances on different networks in east-west traffic. Most often, local ID would be domain name, email address, etc. If this is not configured then the external IP address will be used as the ID.

### Request Example

```
{
  "ipsec_site_connection": {
    "mtu": "2000"
  }
}
```

### Response Parameters

Name	In	Type	Description
auth_mode (Optional)	body	string	The authentication mode. A valid value is psk, which is the default.
ikepolicy_id	body	string	The ID of the IKE policy.
vpnservice_id	body	string	The ID of the VPN service.
local_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private subnets for the local side of the connection. You must specify this parameter with the peer_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
peer_address	body	string	The peer gateway public IPv4 or IPv6 address or FQDN.
id (Optional)	body	string	The ID of the IPSec site-to-site connection.
ipsec_site_connection	body	object	An ipsec_site_connection object.
route_mode (Optional)	body	string	The route mode. A valid value is static, which is the default.

Name	In	Type	Description
ipsecpolicy_id	body	string	The ID of the IPsec policy.
peer_id	body	string	The peer router identity for authentication. A valid value is an IPv4 address, IPv6 address, e-mail address, key ID, or FQDN. Typically, this value matches the peer_address value.
status	body	string	Indicates whether the IPsec connection is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
psk	body	string	The pre-shared key. A valid value is any string.
description	body	string	A human-readable description for the resource.
initiator (Optional)	body	string	Indicates whether this VPN can only respond to connections or both respond to and initiate connections. A valid value is response- only or bi-directional. Default is bi-directional.
peer_cidrs (Optional)	body	array	(Deprecated) Unique list of valid peer private CIDRs in the form < net_address > / < prefix > .
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
interval (Optional)	body	integer	The dead peer detection (DPD) interval, in seconds. A valid value is a positive integer. Default is 30.
mtu	body	integer	The maximum transmission unit (MTU) value to address fragmentation. Minimum value is 68 for IPv4, and 1280 for IPv6.
peer_ep_group_id (Optional)	body	string	The ID for the endpoint group that contains private CIDRs in the form < net_address > / < prefix > for the peer side of the connection. You must specify this parameter with the local_ep_group_id parameter unless in backward-compatible mode where peer_cidrs is provided with a subnet_id for the VPN service.
dpd (Optional)	body	object	A dictionary with dead peer detection (DPD) protocol controls.
timeout	body	integer	The dead peer detection (DPD) timeout in seconds. A valid value is a positive integer that

Name	In	Type	Description
			is greater than the DPD interval value. Default is 120.
action	body	string	The dead peer detection (DPD) action. A valid value is clear, hold, restart, disabled, or restart-by-peer. Default value is hold.
local_id (Optional)	body	string	An ID to be used instead of the external IP address for a virtual router used in traffic between instances on different networks in east-west traffic. Most often, local ID would be domain name, email address, etc. If this is not configured then the external IP address will be used as the ID.

## Response Example

```
{
  "ipsec_site_connection": {
    "status": "DOWN",
    "psk": "secret",
    "initiator": "bi-directional",
    "name": "vpnconnection1",
    "admin_state_up": true,
    "project_id": "10039663455a446d8ba2cbb058b0f578",
    "tenant_id": "10039663455a446d8ba2cbb058b0f578",
    "auth_mode": "psk",
    "peer_cidrs": [],
    "mtu": 2000,
    "peer_ep_group_id": "9ad5a7e0-6dac-41b4-b20d-a7b8645fddf1",
    "ikepolicy_id": "9b00d6b0-6c93-4ca5-9747-b8ade7bb514f",
    "vpnservice_id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "dpd": {
      "action": "hold",
      "interval": 30,
      "timeout": 120
    },
    "route_mode": "static",
    "ipsecpolicy_id": "e6e23d0c-9519-4d52-8ea4-5b1f96d857b1",
    "local_ep_group_id": "3e1815dd-e212-43d0-8f13-b494fa553e68",
    "peer_address": "172.24.4.233",
    "peer_id": "172.24.4.233",
    "id": "851f280f-5639-4ea3-81aa-e298525ab74b",
    "description": "New description"
  }
}
```

## DELETE

/v2.0/vpn/ipsec-site-connections/{connection\_id}

## Remove IPSec connection

Removes an IPSec connection.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
connection_id (Optional)	path	string	The ID of the IPsec site-to-site connection.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/vpn/endpoint-groups

## List VPN endpoint groups

Lists VPN endpoint groups.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
endpoints	body	array	List of endpoints of the same type, for the endpoint group. The values will depend on type.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for

Name	In	Type	Description
			the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
id	body	string	The ID of the VPN endpoint group.

## Response Example

```
{
  "endpoint_groups": [
    {
      "description": "",
      "project_id": "4ad57e7ce0b24fca8f12b9834d91079d",
      "tenant_id": "4ad57e7ce0b24fca8f12b9834d91079d",
      "endpoints": [
        "a3da778c-adfb-46db-88b3-d2ce53290a89"
      ],
      "type": "subnet",
      "id": "6bf34c7c-864c-4948-a6d4-db791669f9d4",
      "name": "locals"
    },
    {
      "description": "",
      "project_id": "4ad57e7ce0b24fca8f12b9834d91079d",
      "tenant_id": "4ad57e7ce0b24fca8f12b9834d91079d",
      "endpoints": [
        "10.2.0.0/24",
        "10.3.0.0/24"
      ],
      "type": "cidr",
      "id": "6ecd9cf3-ca64-46c7-863f-f2eb1b9e838a",
      "name": "peers"
    }
  ]
}
```

**POST**

/v2.0/vpn/endpoint-groups

## Create VPN endpoint group

Creates a VPN endpoint group.

The endpoint group contains one or more endpoints of a specific type that you can use to create a VPN connections.

Normal response codes: 201

Error response codes: 400, 401



## Request

Name	In	Type	Description
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
endpoints	body	array	List of endpoints of the same type, for the endpoint group. The values will depend on type.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
description	body	string	A human-readable description for the resource.
name	body	string	Human-readable name of the resource.

## Request Example

```
{
  "endpoint_group": {
    "endpoints": [
      "10.2.0.0/24",
      "10.3.0.0/24"
    ],
    "type": "cidr",
    "name": "peers"
  }
}
```

## Response Parameters

Name	In	Type	Description
endpoints	body	array	List of endpoints of the same type, for the endpoint group. The values will depend on type.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
id	body	string	The ID of the VPN endpoint group.
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "endpoint_group": {
    "description": "",
    "project_id": "4ad57e7ce0b24fca8f12b9834d91079d",
    "tenant_id": "4ad57e7ce0b24fca8f12b9834d91079d",
    "endpoints": [
      "10.2.0.0/24",
      "10.3.0.0/24"
    ],
    "type": "cidr",
    "id": "6ecd9cf3-ca64-46c7-863f-f2eb1b9e838a",
    "name": "peers"
  }
}
```

**GET**

/v2.0/vpn/endpoint-groups/{endpoint\_group\_id}

## Show VPN endpoint group

Shows details for a VPN endpoint group.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
endpoint_group_id (Optional)	path	string	The ID of the VPN endpoint group.

## Response Parameters

Name	In	Type	Description
endpoints	body	array	List of endpoints of the same type, for the endpoint group. The values will depend on type.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
id	body	string	The ID of the VPN endpoint group.
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "endpoint_group": {
    "description": "",
    "project_id": "4ad57e7ce0b24fca8f12b9834d91079d",
    "tenant_id": "4ad57e7ce0b24fca8f12b9834d91079d",
    "endpoints": [
      "10.2.0.0/24",
      "10.3.0.0/24"
    ],
    "type": "cidr",
    "id": "6ecd9cf3-ca64-46c7-863f-f2eb1b9e838a",
    "name": "peers"
  }
}
```

**PUT**

/v2.0/vpn/endpoint-groups/{endpoint\_group\_id}

## Update VPN endpoint group

Updates settings for a VPN endpoint group.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
name	body	string	Human-readable name of the resource.
endpoint_group_id (Optional)	path	string	The ID of the VPN endpoint group.

## Request Example

```
{
  "endpoint_group": {
    "description": "New description"
  }
}
```

## Response Parameters

Name	In	Type	Description
endpoints	body	array	List of endpoints of the same type, for the endpoint group. The values will depend on type.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.

Name	In	Type	Description
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
id	body	string	The ID of the VPN endpoint group.
name	body	string	Human-readable name of the resource.

### Response Example

```
{
  "endpoint_group": {
    "description": "New description",
    "project_id": "4ad57e7ce0b24fca8f12b9834d91079d",
    "tenant_id": "4ad57e7ce0b24fca8f12b9834d91079d",
    "endpoints": [
      "10.2.0.0/24",
      "10.3.0.0/24"
    ],
    "type": "cidr",
    "id": "6ecd9cf3-ca64-46c7-863f-f2eb1b9e838a",
    "name": "peers"
  }
}
```

### DELETE

/v2.0/vpn/endpoint-groups/{endpoint\_group\_id}

### Remove VPN endpoint group

Removes a VPN endpoint group.

Normal response codes: 204

Error response codes: 401, 404, 409

### Request

Name	In	Type	Description
endpoint_group_id (Optional)	path	string	The ID of the VPN endpoint group.

### Response

There is no body content for the response of a successful DELETE request.

### GET

/v2.0/vpn/vpnservices

### List VPN services

Lists all VPN services.

The list might be empty.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
vpnservices	body	array	A list of VPN service objects.
router_id	path	string	The ID of the router.
status	body	string	Indicates whether IPsec VPN service is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
name	body	string	Human-readable name of the resource.
external_v6_ip	body	string	Read-only external (public) IPv6 address that is used for the VPN service. The VPN plugin sets this address if an IPv6 interface is available.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
subnet_id (Optional)	body	string	If you specify only a subnet UUID, OpenStack Networking allocates an available IP from that subnet to the port. If you specify both a subnet UUID and an IP address, OpenStack Networking tries to allocate the address to the port.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.

Name	In	Type	Description
external_v4_ip	body	string	Read-only external (public) IPv4 address that is used for the VPN service. The VPN plugin sets this address if an IPv4 interface is available.
id	body	string	The ID of the VPN service.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "vpnservices": [
    {
      "router_id": "66e3b16c-8ce5-40fb-bb49-ab6d8dc3f2aa",
      "status": "PENDING_CREATE",
      "name": "myservice",
      "external_v6_ip": "2001:db8::1",
      "admin_state_up": true,
      "subnet_id": null,
      "project_id": "10039663455a446d8ba2cbb058b0f578",
      "tenant_id": "10039663455a446d8ba2cbb058b0f578",
      "external_v4_ip": "172.32.1.11",
      "id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
      "description": ""
    }
  ]
}
```

## POST

/v2.0/vpn/vpnservices

## Create VPN service

Creates a VPN service.

The service is associated with a router. After you create the service, it can contain multiple VPN connections.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
vpnservice	body	object	A vpnservice object.
router_id	path	string	The ID of the router.
description	body	string	A human-readable description for the resource.
admin_state_up	body	boolean	The administrative state of the resource,

Name	In	Type	Description
			which is up (true) or down (false).
subnet_id (Optional)	body	string	If you specify only a subnet UUID, OpenStack Networking allocates an available IP from that subnet to the port. If you specify both a subnet UUID and an IP address, OpenStack Networking tries to allocate the address to the port.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
name	body	string	Human-readable name of the resource.

### Request Example

```
{
  "vpnservice": {
    "subnet_id": null,
    "router_id": "66e3b16c-8ce5-40fb-bb49-ab6d8dc3f2aa",
    "name": "myservice",
    "admin_state_up": true
  }
}
```

### Response Parameters

Name	In	Type	Description
vpnservice	body	object	A vpnservice object.
router_id	path	string	The ID of the router.
status	body	string	Indicates whether IPsec VPN service is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
name	body	string	Human-readable name of the resource.
external_v6_ip	body	string	Read-only external (public) IPv6 address that is used for the VPN service. The VPN plugin sets this address if an IPv6 interface is available.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
subnet_id (Optional)	body	string	If you specify only a subnet UUID, OpenStack Networking allocates an available IP from that subnet to the port. If you

Name	In	Type	Description
			specify both a subnet UUID and an IP address, OpenStack Networking tries to allocate the address to the port.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
external_v4_ip	body	string	Read-only external (public) IPv4 address that is used for the VPN service. The VPN plugin sets this address if an IPv4 interface is available.
id	body	string	The ID of the VPN service.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "vpnservice": {
    "router_id": "66e3b16c-8ce5-40fb-bb49-ab6d8dc3f2aa",
    "status": "PENDING_CREATE",
    "name": "myservice",
    "external_v6_ip": "2001:db8::1",
    "admin_state_up": true,
    "subnet_id": null,
    "project_id": "10039663455a446d8ba2cbb058b0f578",
    "tenant_id": "10039663455a446d8ba2cbb058b0f578",
    "external_v4_ip": "172.32.1.11",
    "id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "description": ""
  }
}
```

**GET**

/v2.0/vpn/vpnservices/{service\_id}

## Show VPN service details

Shows details for a VPN service.

If the user is not an administrative user and the VPN service object does not belong to the tenant account for the user, the operation returns the Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404



## Request

Name	In	Type	Description
service_id (Optional)	path	string	The ID of the VPN service.

## Response Parameters

Name	In	Type	Description
vpnservice	body	object	A vpnservice object.
router_id	path	string	The ID of the router.
status	body	string	Indicates whether IPsec VPN service is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
name	body	string	Human-readable name of the resource.
external_v6_ip	body	string	Read-only external (public) IPv6 address that is used for the VPN service. The VPN plugin sets this address if an IPv6 interface is available.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
subnet_id (Optional)	body	string	If you specify only a subnet UUID, OpenStack Networking allocates an available IP from that subnet to the port. If you specify both a subnet UUID and an IP address, OpenStack Networking tries to allocate the address to the port.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
external_v4_ip	body	string	Read-only external (public) IPv4 address that is used for the VPN service. The VPN plugin sets this address if an IPv4 interface is available.
id	body	string	The ID of the VPN service.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "vpnservice": {
    "router_id": "66e3b16c-8ce5-40fb-bb49-ab6d8dc3f2aa",
    "status": "PENDING_CREATE",
    "name": "myservice",
    "external_v6_ip": "2001:db8::1",
    "admin_state_up": true,
    "subnet_id": null,
    "project_id": "10039663455a446d8ba2cbb058b0f578",
    "tenant_id": "10039663455a446d8ba2cbb058b0f578",
    "external_v4_ip": "172.32.1.11",
    "id": "5c561d9d-eaea-45f6-ae3e-08d1a7080828",
    "description": ""
  }
}
```

**PUT**

/v2.0/vpn/vpnservices/{service\_id}

## Update VPN service

Updates a VPN service.

Updates the attributes of a VPN service. You cannot update a service with a PENDING\_\* status.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
vpnservice	body	object	A vpnservice object.
description	body	string	A human-readable description for the resource.
name	body	string	Human-readable name of the resource.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
service_id (Optional)	path	string	The ID of the VPN service.

## Request Example

```
{
  "vpnservice": {
    "description": "Updated description"
  }
}
```

## Response Parameters

Name	In	Type	Description
vpnservice	body	object	A vpnservice object.
router_id	path	string	The ID of the router.
status	body	string	Indicates whether IPsec VPN service is currently operational. Values are ACTIVE, DOWN, BUILD, ERROR, PENDING_CREATE, PENDING_UPDATE, or PENDING_DELETE.
name	body	string	Human-readable name of the resource.
external_v6_ip	body	string	Read-only external (public) IPv6 address that is used for the VPN service. The VPN plugin sets this address if an IPv6 interface is available.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
subnet_id (Optional)	body	string	If you specify only a subnet UUID, OpenStack Networking allocates an available IP from that subnet to the port. If you specify both a subnet UUID and an IP address, OpenStack Networking tries to allocate the address to the port.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
external_v4_ip	body	string	Read-only external (public) IPv4 address that is used for the VPN service. The VPN plugin sets this address if an IPv4 interface is available.
id	body	string	The ID of the VPN service.
description	body	string	A human-readable description for the resource.

## Response Example

```
{
  "vpnservice": {
    "router_id": "881b7b30-4efb-407e-a162-5630a7af3595",
    "status": "ACTIVE",
    "name": "myvpn",
    "admin_state_up": true,
    "subnet_id": null,
  }
}
```

```
    "project_id": "26de9cd6cae94c8cb9f79d660d628e1f",
    "tenant_id": "26de9cd6cae94c8cb9f79d660d628e1f",
    "id": "41bfef97-af4e-4f6b-a5d3-4678859d2485",
    "description": "Updated description"
  }
}
```

#### DELETE

/v2.0/vpn/vpnservices/{service\_id}

### Remove VPN service

Removes a VPN service.

If the service has connections, the request is rejected.

Normal response codes: 204

Error response codes: 401, 404, 409

#### Request

Name	In	Type	Description
service_id (Optional)	path	string	The ID of the VPN service.

#### Response

There is no body content for the response of a successful DELETE request.

## Resource Management

### Networking Flavors Framework v2.0 (CURRENT) (flavor, service\_profile)

Extension that allows user selection of operator-curated flavors during resource creation.

Users can check if flavor available by performing a GET on the /v2.0/extensions/flavors. If it is unavailable, there is an 404 error response (itemNotFound). Refer [Networking extensions](#) for more details. for more information

#### GET

/v2.0/flavors

### List flavors

Lists all flavors visible to the project.

The list can be empty.

Standard query parameters are supported on the URI. Use the `fields` query

parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#). If Neutron configuration supports pagination by overriding `allow_pagination = false`, the `marker` query parameter can set the last element id the client has seen and `limit` set the maximum number of items to return. If Neutron configuration has `allow_sorting = true`, `sort_key` and `sort_dir` pairs can be used where sort direction is 'asc' or 'desc'.

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
flavors	body	array	A list of flavor objects.
id	body	string	The ID of the flavor.
service_type	body	string	Service type for the flavor. Example: <code>LOADBALANCERV2</code> .
name	body	string	Name of the flavor.
description	body	string	The human-readable description for the flavor.
enabled	body	boolean	Indicates whether the flavor is enabled or not. Default is <code>true</code> .
service_profiles	body	array	Service profile UUIDs associated with this flavor.

## Response Example

```
{
  "flavors": [
    {
      "description": "",
      "enabled": true,
      "service_profiles": [],
      "service_type": "LOADBALANCERV2",
```

```
    "id": "f7b14d9a-b0dc-4fbe-bb14-a0f4970a69e0",  
    "name": "dummy"  
  }  
}
```

**POST**

/v2.0/flavors

## Create flavor

Creates a flavor.

This operation establishes a new flavor.

The `service_type` to which the flavor applies is a required parameter. The corresponding service plugin must have been activated as part of the configuration. See [Service providers](#) for how to see currently loaded service types. Additionally the service plugin needs to support the use of flavors. For example, the `LOADBALANCERV2` service type using the `LBaaSv2` API currently supports Neutron service flavors.

Creation currently limited to administrators. Other users will receive a `Forbidden 403` response code with a response body `NeutronError` message expressing that creation is disallowed by policy.

Until one or more service profiles are associated with the flavor by the operator, attempts to use the flavor during resource creations will currently return a `Not Found 404` with a response body that indicates no service profile could be found.

If the API cannot fulfill the request due to insufficient data or data that is not valid, the service returns the `HTTP Bad Request (400)` response code with information about the failure in the response body. Validation errors require that you correct the error and submit the request again.

Normal response codes: 201

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
flavor	body	object	A flavor object.
service_type	body	string	Service type for the flavor. Example: <code>LOADBALANCERV2</code> .
enabled (Optional)	body	boolean	Indicates whether the flavor is enabled or not. Default is <code>true</code> .
description (Optional)	body	string	The human-readable description for the flavor.
name (Optional)	body	string	Name of the flavor.

## Request Example

```
{
  "flavor": {
    "service_type": "LOADBALANCERV2",
    "enabled": true,
    "name": "dummy",
    "description": "Dummy flavor"
  }
}
```

## Response Parameters

Name	In	Type	Description
flavor	body	object	A flavor object.
id	body	string	The ID of the flavor.
service_type	body	string	Service type for the flavor. Example: LOADBALANCERV2.
name	body	string	Name of the flavor.
description	body	string	The human-readable description for the flavor.
enabled	body	boolean	Indicates whether the flavor is enabled or not. Default is true.
service_profiles	body	array	Service profile UUIDs associated with this flavor.

**GET**

/v2.0/flavors/{flavor\_id}

## Show flavor details

Shows details for a flavor.

This operation returns a flavor object by ID. If you are not an administrative user and the flavor object is not visible to your project account, the service returns the HTTP Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
flavor_id	path	string	The UUID of the flavor.

## Response Parameters

Name	In	Type	Description
flavor	body	object	A flavor object.
id	body	string	The ID of the flavor.
service_type	body	string	Service type for the flavor. Example: LOADBALANCERV2.
name	body	string	Name of the flavor.
description	body	string	The human-readable description for the flavor.
enabled	body	boolean	Indicates whether the flavor is enabled or not. Default is true.
service_profiles	body	array	Service profile UUIDs associated with this flavor.

## Response Example

```
{
  "flavor": {
    "description": "",
    "enabled": true,
    "service_profiles": [],
    "service_type": "LOADBALANCERV2",
    "id": "f7b14d9a-b0dc-4fbe-bb14-a0f4970a69e0",
    "name": "dummy"
  }
}
```

**PUT**

/v2.0/flavors/{flavor\_id}

## Update flavor

Updates a flavor.

The service\_type cannot be updated as there may be associated service profiles and consumers depending on the value.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
flavor_id	path	string	The UUID of the flavor.
flavor	body	object	A flavor object.
name (Optional)	body	string	Name of the flavor.



Name	In	Type	Description
description (Optional)	body	string	The human-readable description for the flavor.
enabled (Optional)	body	boolean	Indicates whether the flavor is enabled or not. Default is true.

### Request Example

```
{
  "flavor": {
    "enabled": false,
    "name": "newname",
    "description": "New description"
  }
}
```

### Response Parameters

Name	In	Type	Description
flavor	body	object	A flavor object.
id	body	string	The ID of the flavor.
service_type	body	string	Service type for the flavor. Example: LOADBALANCERV2.
name	body	string	Name of the flavor.
description	body	string	The human-readable description for the flavor.
enabled	body	boolean	Indicates whether the flavor is enabled or not. Default is true.
service_profiles	body	array	Service profile UUIDs associated with this flavor.

### Response Example

```
{
  "flavor": {
    "description": "New description",
    "enabled": false,
    "service_profiles": [],
    "service_type": "LOADBALANCERV2",
    "id": "7fc0581b-4509-49e1-90eb-c953c877fa4c",
    "name": "newname"
  }
}
```

### DELETE

/v2.0/flavors/{flavor\_id}

### Delete flavor

Deletes a flavor.

Normal response codes: 204

Error response codes: 401, 403, 404

#### Request

Name	In	Type	Description
flavor_id	path	string	The UUID of the flavor.

#### Response

No body content is returned on a successful DELETE.

#### POST

/v2.0/flavors/{flavor\_id}/service\_profiles

### Associate flavor with a service profile

Associate a flavor with a service profile.

A flavor can be associated with more than one profile.

Will return 409 Conflict if association already exists.

Normal response codes: 201

Error response codes: 400, 401, 403, 404, 409

#### Request

Name	In	Type	Description
flavor_id	path	string	The UUID of the flavor.
service_profile	body	object	A service_profile object.
id	body	string	The UUID of the service profile.

#### Request Example

```
{
  "service_profile": {
    "id": "4e5b9191-ffbe-4f7a-b112-2db98232fd32"
  }
}
```

#### Response Parameters

Name	In	Type	Description
service_profile	body	object	A service_profile object.
id	body	string	The ID of the resource.

#### DELETE

/v2.0/flavors/{flavor\_id}/service\_profiles/{profile\_id}

### Disassociate a flavor.

Disassociate a flavor from a service profile.

Normal response codes: 204

Error response codes: 401, 403, 404

#### Request

Name	In	Type	Description
profile_id	path	string	The UUID of the service profile.
flavor_id	path	string	The UUID of the flavor.

#### Response

No body content is returned on a successful disassociation.

**GET**

/v2.0/service\_profiles

### List service profiles

Lists all service profiles visible for the tenant account.

The list can be empty.

Standard query parameters are supported on the URI.

Normal response codes: 200

Error response codes: 401

#### Request

#### Response Parameters

Name	In	Type	Description
service_profiles	body	array	Service profile UUIDs associated with this flavor.
id	body	string	The UUID of the service profile.
enabled	body	boolean	Indicates whether this service profile is enabled or not. Default is true.
driver	body	string	Provider driver to use for this profile. Example: <code>neutron_lbaas.drivers.octavia.driver.OctaviaDriver</code>
description	body	string	The human-readable description for the service profile.

Name	In	Type	Description
metainfo	body	string	JSON-formatted meta information of the service profile.

## Response Example

```
{
  "service_profiles": [
    {
      "id": "4e5b9191-ffbe-4f7a-b112-2db98232fd32",
      "enabled": true,
      "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
      "description": "",
      "metainfo": "{}"
    },
    {
      "id": "684322c5-703a-48a2-8138-34b99942a7ef",
      "enabled": true,
      "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
      "description": "",
      "metainfo": "{}"
    }
  ]
}
```

**POST**

/v2.0/service\_profiles

## Create service profile

Creates a service profile.

This operation establishes a new service profile that can be associated with one or more flavors.

Either metadata or a driver is required.

If a driver is specified but does not exist, call will return a Not found 404 error with the response body explaining that the driver could not be found.

Creation currently limited to administrators. Other users will receive a Forbidden 403 response code with a response body NeutronError message expressing that creation is disallowed by policy.

If the API cannot fulfill the request due to insufficient data or data that is not valid, the service returns the HTTP Bad Request (400) response code with information about the failure in the response body. Validation errors require that you correct the error and submit the request again.

Normal response codes: 201

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
service_profile	body	object	A service_profile object.
description (Optional)	body	string	The human-readable description for the service profile.
metainfo (Optional)	body	string	JSON-formatted meta information of the service profile.
enabled (Optional)	body	boolean	Indicates whether this service profile is enabled or not. Default is true.
driver (Optional)	body	string	Provider driver to use for this profile. Example: neutron_lbaas.drivers.octavia.driver.OctaviaDriver

## Request Example

```
{
  "service_profile": {
    "enabled": "true",
    "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
    "description": "Dummy profile",
    "metainfo": "{\"foo\": 'bar'}"
  }
}
```

## Response Parameters

Name	In	Type	Description
service_profile	body	object	A service_profile object.
id	body	string	The UUID of the service profile.
enabled	body	boolean	Indicates whether this service profile is enabled or not. Default is true.
driver	body	string	Provider driver to use for this profile. Example: neutron_lbaas.drivers.octavia.driver.OctaviaDriver
description	body	string	The human-readable description for the service profile.
metainfo	body	string	JSON-formatted meta information of the service profile.

## Response Example

```
{
  "service_profile": {
    "enabled": true,
    "metainfo": "{\"foo\": 'bar'}",
    "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
    "id": "7c793e5f-9b64-44e0-8b1f-902e59c85a01",
    "description": "Dummy profile"
  }
}
```

GET

/v2.0/service\_profiles/{profile\_id}

## Show service profile details

Shows details for a service profile.

This operation returns a service profile object by ID. If you are not an administrative user and the object is not visible to your tenant account, the service returns the HTTP Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
profile_id	path	string	The UUID of the service profile.

### Response Parameters

Name	In	Type	Description
service_profile	body	object	A service_profile object.
id	body	string	The UUID of the service profile.
enabled	body	boolean	Indicates whether this service profile is enabled or not. Default is true.
driver	body	string	Provider driver to use for this profile. Example: neutron_lbaas.drivers.octavia.driver.OctaviaDriver
description	body	string	The human-readable description for the service profile.
metainfo	body	string	JSON-formatted meta information of the service profile.

### Response Example

```
{
  "service_profile": {
    "enabled": true,
    "metainfo": "{\"foo\": 'bar'}",
    "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
    "id": "7c793e5f-9b64-44e0-8b1f-902e59c85a01",
    "description": "Dummy profile"
  }
}
```

PUT

/v2.0/service\_profiles/{profile\_id}

## Update service profile

Updates a service profile.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

### Request

Name	In	Type	Description
profile_id	path	string	The UUID of the service profile.
service_profile	body	object	A service_profile object.
enabled (Optional)	body	boolean	Indicates whether this service profile is enabled or not. Default is true.
driver (Optional)	body	string	Provider driver to use for this profile. Example: neutron_lbaas.drivers.octavia.driver.OctaviaDriver
description (Optional)	body	string	The human-readable description for the service profile.
metainfo (Optional)	body	string	JSON-formatted meta information of the service profile.

### Request Example

```
{
  "service_profile": {
    "enabled": false,
    "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
    "description": "New description",
    "metainfo": '{"new': 'info'}"
  }
}
```

### Response Parameters

Name	In	Type	Description
service_profile	body	object	A service_profile object.
id	body	string	The UUID of the service profile.
enabled	body	boolean	Indicates whether this service profile is enabled or not. Default is true.
driver	body	string	Provider driver to use for this profile. Example: neutron_lbaas.drivers.octavia.driver.OctaviaDriver
description	body	string	The human-readable description for the service profile.
metainfo	body	string	JSON-formatted meta information of the service profile.

### Response Example

```
{
```

```

    "service_profile": {
      "enabled": false,
      "metainfo": '{"new': 'info'}',
      "driver": "neutron_lbaas.drivers.octavia.driver.OctaviaDriver",
      "id": "7c793e5f-9b64-44e0-8b1f-902e59c85a01",
      "description": "New description"
    }
  }
}

```

## DELETE

/v2.0/service\_profiles/{profile\_id}

### Delete service profile

Deletes a service profile.

Attempting to delete a service profile that is currently associated with a flavor will return a Conflict 409 with a response body containing an in use message.

Either metadata or a driver is required.

Normal response codes: 204

Error response codes: 401, 403, 404, 409

#### Request

Name	In	Type	Description
profile_id	path	string	The UUID of the service profile.

#### Response

No body content is returned on a successful DELETE.

## Metering labels and rules (metering-labels, metering-label-rules)

Creates, modifies, and deletes OpenStack Layer3 metering labels and rules.

## GET

/v2.0/metering/metering-labels

### List metering labels

Lists all L3 metering labels that belong to the project.

The list shows the ID for each metering label.

Use the fields query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200



## Error response codes: 401

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
metering_labels	body	array	A list of <code>metering_label</code> objects.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
shared	body	boolean	Indicates whether this metering label is shared across all projects.
id	body	string	The ID of the metering label.
name	body	string	Human-readable name of the resource.

### Response Example

```
{
  "metering_labels": [
    {
      "project_id": "45345b0ee1ea477fac0f541b2cb79cd4",
      "tenant_id": "45345b0ee1ea477fac0f541b2cb79cd4",
      "description": "label1 description",
      "name": "label1",
      "id": "a6700594-5b7a-4105-8bfe-723b346ce866",
      "shared": false
    },
    {
      "project_id": "45345b0ee1ea477fac0f541b2cb79cd4",
      "tenant_id": "45345b0ee1ea477fac0f541b2cb79cd4",
      "description": "label2 description",
      "name": "label2",
      "id": "e131d186-b02d-4c0b-83d5-0c0725c4f812",
      "shared": false
    }
  ]
}
```

**POST**

/v2.0/metering/metering-labels

## Create metering label

Creates an L3 metering label.

Normal response codes: 201

Error response codes: 400, 401, 403

### Request

Name	In	Type	Description
metering_label	body	object	A metering_label object.
shared	body	boolean	Indicates whether this metering label is shared across all projects.
description	body	string	A human-readable description for the resource.
name	body	string	Human-readable name of the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.

### Request Example

```
{
  "metering_label": {
    "name": "label1",
    "description": "description of label1"
  }
}
```

### Response Parameters

Name	In	Type	Description
metering_label	body	object	A metering_label object.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
shared	body	boolean	Indicates whether this metering label is shared across all projects.

Name	In	Type	Description
id	body	string	The ID of the metering label.
name	body	string	Human-readable name of the resource.

**GET**

/v2.0/metering/metering-labels/{metering\_label\_id}

## Show metering label details

Shows details for a metering label.

Normal response codes: 200

Error response codes: 401, 404

### Request

Name	In	Type	Description
metering_label_id	path	string	The ID of the metering label.

### Request Example

```
GET /v2.0/metering/metering-labels/a6700594-5b7a-4105-8bfe-723b346ce866 HTTP/1.1
Host: controlnode:9696
User-Agent: python-neutronclient
Content-Type: application/json
Accept: application/json
X-Auth-Token: c52a1b304fec4ca0ac85dc1741eec6e2
```

### Response Parameters

Name	In	Type	Description
metering_label	body	object	A metering_label object.
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
shared	body	boolean	Indicates whether this metering label is shared across all projects.
id	body	string	The ID of the metering label.
name	body	string	Human-readable name of the resource.

### Response Example

```
{
  "metering_label": {
```

```
    "project_id": "45345b0ee1ea477fac0f541b2cb79cd4",
    "tenant_id": "45345b0ee1ea477fac0f541b2cb79cd4",
    "description": "label1 description",
    "name": "label1",
    "id": "a6700594-5b7a-4105-8bfe-723b346ce866",
    "shared": false
  }
}
```

## DELETE

/v2.0/metering/metering-labels/{metering\_label\_id}

### Delete metering label

Deletes an L3 metering label.

Normal response codes: 204

Error response codes: 401, 404

#### Request

Name	In	Type	Description
metering_label_id	path	string	The ID of the metering label.

#### Request Example

```
DELETE /v2.0/metering/metering-labels/a6700594-5b7a-4105-8bfe-723b346ce866 HTTP/1.1
Host: controlnode:9696
User-Agent: python-neutronclient
Content-Type: application/json
Accept: application/json
X-Auth-Token: c52a1b304fec4ca0ac85dc1741eec6e2
```

#### Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/metering/metering-label-rules

### List metering label rules

Lists a summary of all L3 metering label rules that belong to the project.

The list shows the ID for each metering label rule.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
metering_label_rules	body	array	A list of <code>metering_label_rule</code> objects.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
excluded (Optional)	body	boolean	Indicates whether to count the traffic of a specific IP address with the <code>remote_ip_prefix</code> value. Default is <code>false</code> .
metering_label_id	body	string	The metering label ID to associate with this metering rule.
id	body	string	The ID of the metering label rule.

## Response Example

```
{
  "metering_label_rules": [
    {
      "remote_ip_prefix": "20.0.0.0/24",
      "direction": "ingress",
      "metering_label_id": "e131d186-b02d-4c0b-83d5-0c0725c4f812",
      "id": "9536641a-7d14-4dc5-afaf-93a973ce0eb8",
      "excluded": false
    },
    {
      "remote_ip_prefix": "10.0.0.0/24",
      "direction": "ingress",
      "metering_label_id": "e131d186-b02d-4c0b-83d5-0c0725c4f812",
      "id": "fffc6fd15-40de-4e7d-b617-34d3f7a93aec",
      "excluded": false
    }
  ]
}
```

**POST**

/v2.0/metering/metering-label-rules

### Create metering label rule

Creates an L3 metering label rule.

Normal response codes: 201

Error response codes: 400, 401, 403, 404, 409

## Request

Name	In	Type	Description
metering_label_rule	body	object	A metering_label_rule object.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
metering_label_id	body	string	The metering label ID to associate with this metering rule.
excluded (Optional)	body	boolean	Indicates whether to count the traffic of a specific IP address with the remote_ip_prefix value. Default is false.

## Request Example

```
{
  "metering_label_rule": {
    "remote_ip_prefix": "10.0.1.0/24",
    "direction": "ingress",
    "metering_label_id": "e131d186-b02d-4c0b-83d5-0c0725c4f812"
  }
}
```

## Response Parameters

Name	In	Type	Description
metering_label_rule	body	object	A metering_label_rule object.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
excluded (Optional)	body	boolean	Indicates whether to count the traffic of a specific IP address with the remote_ip_prefix value. Default is false.

Name	In	Type	Description
metering_label_id	body	string	The metering label ID to associate with this metering rule.
id	body	string	The ID of the metering label rule.

## Response Example

```
{
  "metering_label_rule": {
    "remote_ip_prefix": "10.0.1.0/24",
    "direction": "ingress",
    "metering_label_id": "e131d186-b02d-4c0b-83d5-0c0725c4f812",
    "id": "00e13b58-b4f2-4579-9c9c-7ac94615f9ae",
    "excluded": false
  }
}
```

**GET**

/v2.0/metering/metering-label-rules/{metering\_label\_rule\_id}

## Show metering label rule details

Shows details for a metering label rule.

The response body shows this information for each metering label rule:

- direction. Either ingress or egress.
- excluded. Either true or false.
- The ID for the metering label rule.
- The remote IP prefix.
- The metering label ID for the metering label with which the rule is associated.

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
metering_label_rule_id	path	string	The ID of the metering label rule.

## Request Example

```
GET /v2.0/metering/metering-label-rules/9536641a-7d14-4dc5-afaf-93a973ce0eb8 HTTP/1.1
Host: controlnode:9696
User-Agent: python-neutronclient
Content-Type: application/json
Accept: application/json
X-Auth-Token: c52a1b304fec4ca0ac85dc1741eec6e2
```

## Response Paramters

Name	In	Type	Description
metering_label_rule	body	object	A metering_label_rule object.
direction	body	string	Ingress or egress, which is the direction in which the metering rule is applied.
remote_ip_prefix	body	string	The remote IP prefix to associate with this metering rule packet.
excluded (Optional)	body	boolean	Indicates whether to count the traffic of a specific IP address with the remote_ip_prefix value. Default is false.
metering_label_id	body	string	The metering label ID to associate with this metering rule.
id	body	string	The ID of the metering label rule.

## Response Example

```
{
  "metering_label_rule": {
    "remote_ip_prefix": "20.0.0.0/24",
    "direction": "ingress",
    "metering_label_id": "e131d186-b02d-4c0b-83d5-0c0725c4f812",
    "id": "9536641a-7d14-4dc5-afaf-93a973ce0eb8",
    "excluded": false
  }
}
```

## DELETE

/v2.0/metering/metering-label-rules/{metering\_label\_rule\_id}

## Delete metering label rule

Deletes an L3 metering label rule.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
metering_label_rule_id	path	string	The ID of the metering label rule.

## Request Example

```
DELETE /v2.0/metering/metering-labels/37b31179-71ee-4f0a-b130-0eeb28e7ede7 HTTP/1.1
Host: controlnode:9696
User-Agent: python-neutronclient
Content-Type: application/json
Accept: application/json
X-Auth-Token: c52a1b304fec4ca0ac85dc1741eec6e2
```



## Response

There is no body content for the response of a successful DELETE request.

## Network IP availability and usage stats

The extension `network-ip-availability` allows users to list and show the network IP usage stats of all networks or of a specified network. By default policy configuration, only administrative users can use this API.

**GET**

`/v2.0/network-ip-availabilities/{network_id}`

### Show Network IP Availability

Shows network IP availability details for a network.

By default policy configuration, only administrative users can retrieve IP availability. Otherwise, Not Found (404) will be returned.

Normal response codes: 200

Error response codes: 401, 404

### Request

Name	In	Type	Description
network_id	path	string	The ID of the network.

### Response Parameters

Name	In	Type	Description
network_ip_availability	body	object	A <code>network_ip_availability</code> object.
network_id	body	string	The ID of the network whose IP availability detail is reported.
network_name	body	string	Human-readable name of the network.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
total_ips	body	integer	The total number of IP addresses in a network.
used_ips	body	integer	The number of used IP addresses of all subnets in a network.
subnet_ip_availability	body	array	A list of dictionaries showing subnet IP availability. It contains

Name	In	Type	Description
			information for every subnet associated to the network.
subnet_id	body	string	The ID of the subnet whose IP availability detail is reported.
subnet_name	body	string	The name of the subnet.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
cidr	body	string	The CIDR of the subnet.

### Response Example

```
{
  "network_ip_availability": {
    "used_ips": 4,
    "subnet_ip_availability": [
      {
        "used_ips": 2,
        "subnet_id": "44e70d00-80a2-4fb1-ab59-6190595ceb61",
        "subnet_name": "private-subnet",
        "ip_version": 4,
        "cidr": "10.0.0.0/24",
        "total_ips": 253
      },
      {
        "used_ips": 2,
        "subnet_id": "a90623df-00e1-4902-a675-40674385d74c",
        "subnet_name": "ipv6-private-subnet",
        "ip_version": 6,
        "cidr": "fdbf:ac66:9be8::/64",
        "total_ips": 18446744073709552000
      }
    ],
    "network_id": "6801d9c8-20e6-4b27-945d-62499f00002e",
    "project_id": "d56d3b8dd6894a508cf41b96b522328c",
    "tenant_id": "d56d3b8dd6894a508cf41b96b522328c",
    "total_ips": 18446744073709552000,
    "network_name": "private"
  }
}
```

**GET**

/v2.0/network-ip-availabilities

### List Network IP Availability

Lists network IP availability of all networks.

By default policy configuration, only administrative users can retrieve IP availabilities. Otherwise, an empty list will be returned.

Normal response codes: 200

Error response codes: 401

Request

## Response Parameters

Name	In	Type	Description
network_ip_availabilities	body	array	The network_ip_availabilities object.
network_id	body	string	The ID of the network whose IP availability detail is reported.
network_name	body	string	Human-readable name of the network.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
total_ips	body	integer	The total number of IP addresses in a network.
used_ips	body	integer	The number of used IP addresses of all subnets in a network.
subnet_ip_availability	body	array	A list of dictionaries showing subnet IP availability. It contains information for every subnet associated to the network.
subnet_id	body	string	The ID of the subnet whose IP availability detail is reported.
subnet_name	body	string	The name of the subnet.
ip_version	body	integer	The IP protocol version. Value is 4 or 6.
cidr	body	string	The CIDR of the subnet.

## Response Example

```
{
  "network_ip_availabilities": [
    {
      "network_id": "4cf895c9-c3d1-489e-b02e-59b5c8976809",
      "network_name": "public",
      "subnet_ip_availability": [
        {
          "cidr": "2001:db8::/64",
          "ip_version": 6,
          "subnet_id": "ca3f46c4-c6ff-4272-9be4-0466f84c6077",
          "subnet_name": "ipv6-public-subnet",
          "total_ips": 18446744073709552000,
          "used_ips": 1
        }
      ],
      "cidr": "172.24.4.0/24",
      "ip_version": 4,
      "subnet_id": "cc02efc1-9d47-46bd-bab6-760919c836b5",
      "subnet_name": "public-subnet",
    }
  ]
}
```

```

        "total_ips": 253,
        "used_ips": 1
    },
    {
        "project_id": "1a02cc95f1734fcc9d3c753818f03002",
        "tenant_id": "1a02cc95f1734fcc9d3c753818f03002",
        "total_ips": 253,
        "used_ips": 2
    },
    {
        "network_id": "6801d9c8-20e6-4b27-945d-62499f00002e",
        "network_name": "private",
        "subnet_ip_availability": [
            {
                "cidr": "10.0.0.0/24",
                "ip_version": 4,
                "subnet_id": "44e70d00-80a2-4fb1-ab59-6190595ceb61",
                "subnet_name": "private-subnet",
                "total_ips": 253,
                "used_ips": 2
            },
            {
                "ip_version": 6,
                "cidr": "fdbf:ac66:9be8::/64",
                "subnet_id": "a90623df-00e1-4902-a675-40674385d74c",
                "subnet_name": "ipv6-private-subnet",
                "total_ips": 18446744073709552000,
                "used_ips": 2
            }
        ],
        "project_id": "d56d3b8dd6894a508cf41b96b522328c",
        "tenant_id": "d56d3b8dd6894a508cf41b96b522328c",
        "total_ips": 18446744073709552000,
        "used_ips": 4
    }
]
}

```

## Quotas extension (quotas)

Lists default quotas, current quotas for projects with non-default quota values, and shows, updates, and resets quotas for a project.

A quota value of -1 means that quota has no limit.

**GET**

/v2.0/quotas

### List quotas for projects with non-default quota values

Lists quotas for projects with non-default quota values.

Normal response codes: 200

Error response codes: 401, 403

## Request

## Response Parameters

Name	In	Type	Description
subnet	body	integer	The number of subnets allowed for each project.
network	body	integer	The number of networks allowed for each project.
floatingip	body	integer	The number of floating IP addresses allowed for each project. A value of -1 means no limit.
subnetpool	body	integer	The number of subnet pools allowed for each project.
quotas	body	array	A list of quota objects.
security_group_rule	body	integer	The number of security group rules allowed for each project.
security_group	body	integer	The number of security groups allowed for each project.
router	body	integer	The number of routers allowed for each project.
rbac_policy	body	integer	The number of role-based access control (RBAC) policies for each project.
port	body	integer	The number of ports allowed for each project.

## Response Example

```
{
  "quotas": [
    {
      "subnet": 10,
      "network": 15,
      "floatingip": 50,
      "project_id": "bab7d5c60cd041a0a36f7c4b6e1dd978",
      "tenant_id": "bab7d5c60cd041a0a36f7c4b6e1dd978",
      "subnetpool": -1,
      "security_group_rule": 100,
      "security_group": 10,
      "router": 10,
      "rbac_policy": -1,
      "port": 50
    }
  ]
}
```

GET

/v2.0/quotas/{project\_id}

## List quotas for a project

Lists quotas for a project.

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
project_id	path	string	The ID of the project.

### Response Parameters

Name	In	Type	Description
subnet	body	integer	The number of subnets allowed for each project.
network	body	integer	The number of networks allowed for each project.
floatingip	body	integer	The number of floating IP addresses allowed for each project. A value of -1 means no limit.
subnetpool	body	integer	The number of subnet pools allowed for each project.
quotas	body	array	A list of quota objects.
security_group_rule	body	integer	The number of security group rules allowed for each project.
security_group	body	integer	The number of security groups allowed for each project.
router	body	integer	The number of routers allowed for each project.
rbac_policy	body	integer	The number of role-based access control (RBAC) policies for each project.
port	body	integer	The number of ports allowed for each project.

### Response Example

```
{  
  "quota": {
```

```

        "subnet": 10,
        "network": 10,
        "floatingip": 50,
        "subnetpool": -1,
        "security_group_rule": 100,
        "security_group": 10,
        "router": 10,
        "rbac_policy": -1,
        "port": 50
    }
}

```

**PUT**

/v2.0/quotas/{project\_id}

## Update quota for a project

Updates quotas for a project. Use when non-default quotas are desired.

Normal response codes: 200

Error response codes: 401, 403, 404

### Request

Name	In	Type	Description
project_id	path	string	The ID of the project.
subnet	body	integer	The number of subnets allowed for each project.
network	body	integer	The number of networks allowed for each project.
floatingip	body	integer	The number of floating IP addresses allowed for each project. A value of -1 means no limit.
subnetpool	body	integer	The number of subnet pools allowed for each project.
quotas	body	array	A list of quota objects.
security_group_rule	body	integer	The number of security group rules allowed for each project.
security_group	body	integer	The number of security groups allowed for each project.
router	body	integer	The number of routers allowed for each project.
rbac_policy	body	integer	The number of role-based access control (RBAC)

Name	In	Type	Description
			policies for each project.
port	body	integer	The number of ports allowed for each project.

### Request Example

```
{
  "quota": {
    "subnet": 10,
    "network": 10,
    "floatingip": 50,
    "subnetpool": -1,
    "security_group_rule": 100,
    "security_group": 10,
    "router": 10,
    "rbac_policy": -1,
    "port": 50
  }
}
```

### Response Parameters

Name	In	Type	Description
subnet	body	integer	The number of subnets allowed for each project.
network	body	integer	The number of networks allowed for each project.
floatingip	body	integer	The number of floating IP addresses allowed for each project. A value of -1 means no limit.
subnetpool	body	integer	The number of subnet pools allowed for each project.
quotas	body	array	A list of quota objects.
security_group_rule	body	integer	The number of security group rules allowed for each project.
security_group	body	integer	The number of security groups allowed for each project.
router	body	integer	The number of routers allowed for each project.
rbac_policy	body	integer	The number of role-based access control (RBAC) policies for each project.
port	body	integer	The number of ports



Name	In	Type	Description
			allowed for each project.

## Response Example

```
{
  "quota": {
    "subnet": 10,
    "network": 15,
    "floatingip": 50,
    "subnetpool": -1,
    "security_group_rule": 100,
    "security_group": 10,
    "router": 10,
    "rbac_policy": -1,
    "port": 50
  }
}
```

## DELETE

/v2.0/quotas/{project\_id}

## Reset quota for a project

Resets quotas to default values for a project.

Normal response codes: 204

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
project_id	path	string	The ID of the project.

## GET

/v2.0/quotas/{project\_id}/default

## List default quotas for a project

Lists default quotas for a project.

Normal response codes: 200

Error response codes: 401, 403, 404

## Request

Name	In	Type	Description
project_id	path	string	The ID of the project.

## Response Parameters

Name	In	Type	Description
quota	body	object	A quota object.
floatingip	body	integer	The number of floating IP addresses allowed for each project. A value of -1 means no limit.
network	body	integer	The number of networks allowed for each project.
port	body	integer	The number of ports allowed for each project.
rbac_policy	body	integer	The number of role-based access control (RBAC) policies for each project.
router	body	integer	The number of routers allowed for each project.
security_group	body	integer	The number of security groups allowed for each project.
security_group_rule	body	integer	The number of security group rules allowed for each project.
subnet	body	integer	The number of subnets allowed for each project.
subnetpool	body	integer	The number of subnet pools allowed for each project.

## Response Example

```
{
  "quota": {
    "subnet": 10,
    "network": 10,
    "floatingip": 50,
    "subnetpool": -1,
    "security_group_rule": 100,
    "security_group": 10,
    "router": 10,
    "rbac_policy": -1,
    "port": 50
  }
}
```

## Service providers

Lists service providers.

**GET**

/v2.0/service-providers

## List service providers

Lists service providers and their associated service types.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
service_providers	body	array	A list of <code>service_provider</code> objects.
service_type	body	string	The service type, which is <code>CORE</code> , <code>DUMMY</code> , <code>FIREWALL</code> , <code>FLAVORS</code> , <code>L3_ROUTER_NAT</code> , <code>LOADBALANCER</code> , <code>LOADBALANCERV2</code> , <code>METERING</code> , <code>QOS</code> , or <code>VPN</code> .
name	body	string	Human-readable name of the resource.
default	body	boolean	Defines whether the provider is the default for the service type. If this value is <code>true</code> , the provider is the default. If this value is <code>false</code> , the provider is not the default.

### Response Example

```
{
  "service_providers": [
    {
      "service_type": "LOADBALANCER",
      "default": true,
      "name": "haproxy"
    }
  ]
}
```

# Tag extension (tags)

Shows details for, updates, and deletes tags.

The maximum number of characters allowed in a tag is 60. If the length is longer than 60, the API returns the HTTP Bad Request (400) response code with 'invalid input for operation' error message.

## Tag Extension

The tag extension allows users to set tags on their networks. This extension supports networks only.

## Enhanced Tag Extension

The tag-ext extension allows users to set tags on their resources. This extension supports subnets, ports, routers, and subnet pools.

**PUT**

/v2.0/{resource\_type}/{resource\_id}/tags

### Replace all tags

Replaces all tags on the resource.

Normal response codes: 200

Error response codes: 400, 401, 404

Request

Name	In	Type	Description
resource_type	path	string	The type of resource which the tag is set on.
resource_id	path	string	The ID of resource which the tag is set on.
tags	body	array	The list of tags on the resource.

Request Example

```
{
  "tags": [
    "red",
    "blue"
  ]
}
```

Response Parameters

Name	In	Type	Description
tags	body	array	The list of tags on the resource.

Name	In	Type	Description

## Response Example

```
{
  "tags": [
    "red",
    "blue"
  ]
}
```

## DELETE

/v2.0/{resource\_type}/{resource\_id}/tags

## Remove all tags

Removes all tags on the resource.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
resource_type	path	string	The type of resource which the tag is set on.
resource_id	path	string	The ID of resource which the tag is set on.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/{resource\_type}/{resource\_id}/tags/{tag}

## Confirm a tag

Confirms a given tag is set on the resource.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
resource_type	path	string	The type of resource which the tag is set on.
resource_id	path	string	The ID of resource which the

Name	In	Type	Description
			tag is set on.
tag	path	string	The name for the tag.

## Response

There is no body content for the response of a successful GET request.

## PUT

/v2.0/{resource\_type}/{resource\_id}/tags/{tag}

## Add a tag

Adds a tag on the resource.

Normal response codes: 201

Error response codes: 401, 404

## Request

Name	In	Type	Description
resource_type	path	string	The type of resource which the tag is set on.
resource_id	path	string	The ID of resource which the tag is set on.
tag	path	string	The name for the tag.

## Response

There is no body content for the response of a successful PUT request.

## GET

/v2.0/{resource\_type}/{resource\_id}/tags

## Obtain Tag List

Obtains the tags for a resource.

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
resource_type	path	string	The type of resource which the tag is set on.
resource_id	path	string	The ID of resource which the

Name	In	Type	Description
			tag is set on.

## Response

Name	In	Type	Description
tags	body	array	The list of tags on the resource.

## Response Example

```
{
  "tags": [
    "red",
    "blue"
  ]
}
```

## DELETE

/v2.0/{resource\_type}/{resource\_id}/tags/{tag}

## Remove a tag

Removes a tag on the resource.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
resource_type	path	string	The type of resource which the tag is set on.
resource_id	path	string	The ID of resource which the tag is set on.
tag	path	string	The name for the tag.

## Response

There is no body content for the response of a successful DELETE request.

# Quality of Service

## QoS policies (qos)

Lists, creates, shows information for, and updates QoS policies.

## GET

/v2.0/qos/policies/{policy\_id}/bandwidth\_limit\_rules/{rule\_id}

## Show bandwidth limit rule details

Shows details for a bandwidth limit rule for a QoS policy.

Normal response codes: 200

Error response codes: 401

Request

Name	In	Type	Description
rule_id (Optional)	path	string	The UUID of the rule.
policy_id (Optional)	path	string	The UUID of the policy.

Response Parameters

Name	In	Type	Description
bandwidth_limit_rule	body	object	A bandwidth_limit_rule object.
max_kbps (Optional)	body	integer	The maximum KBPS value. If you specify this value, must be greater than 0. Default is null.
id	body	string	The ID of the resource.
max_burst_kbps (Optional)	body	integer	The burst over the maximum KBPS value. Default is 0
policy_id (Optional)	path	string	The UUID of the policy.

Response Example

```
{
  "bandwidth_limit_rule": {
    "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c793",
    "max_kbps": 10000,
    "max_burst_kbps": 0
  }
}
```

**PUT**

/v2.0/qos/policies/{policy\_id}/bandwidth\_limit\_rules/{rule\_id}

## Update bandwidth limit rule

Updates a bandwidth limit rule for a QoS policy.

If the request is valid, the service returns the Accepted (202) response code.

Normal response codes: 200



Error response codes: 400, 401, 413

## Request

Name	In	Type	Description
bandwidth_limit_rule	body	object	A bandwidth_limit_rule object.
max_kbps (Optional)	body	integer	The maximum KBPS value. If you specify this value, must be greater than 0. Default is null.
max_burst_kbps (Optional)	body	integer	The burst over the maximum KBPS value. Default is 0
policy_id (Optional)	path	string	The UUID of the policy.
rule_id (Optional)	path	string	The UUID of the rule.
policy_id (Optional)	path	string	The UUID of the policy.

## Request Example

```
{
  "bandwidth_limit_rule": {
    "max_kbps": "10000"
  }
}
```

## Response Parameters

Name	In	Type	Description
bandwidth_limit_rule	body	object	A bandwidth_limit_rule object.
max_kbps (Optional)	body	integer	The maximum KBPS value. If you specify this value, must be greater than 0. Default is null.
id	body	string	The ID of the resource.
max_burst_kbps (Optional)	body	integer	The burst over the maximum KBPS value. Default is 0
policy_id (Optional)	path	string	The UUID of the policy.

## Response Example

```
{
  "bandwidth_limit_rule": {
    "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c793",
    "max_kbps": 10000,
    "max_burst_kbps": 0
  }
}
```

**DELETE**

/v2.0/qos/policies/{policy\_id}/bandwidth\_limit\_rules/{rule\_id}

## Delete bandwidth limit rule

Deletes a bandwidth limit rule for a QoS policy.

Normal response codes: 204

Error response codes: 400, 401, 413

Request

Name	In	Type	Description
rule_id (Optional)	path	string	The UUID of the rule.
policy_id (Optional)	path	string	The UUID of the policy.

GET

/v2.0/qos/policies

## List QoS policies

Lists all QoS policies associated with your project.

The list might be empty.

Normal response codes: 200

Error response codes: 401

Request

Response Parameters

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
policy_id (Optional)	path	string	The UUID of the policy.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
policies	body	array	A list of QoS policy objects.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
id	body	string	The ID of the resource.
rules (Optional)	body	array	A set of zero or more policy rules.
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "policies": [
    {
      "project_id": "8d4c70a21fed4aeba121a1a429ba0d04",
      "tenant_id": "8d4c70a21fed4aeba121a1a429ba0d04",
      "id": "46ebaec0-0570-43ac-82f6-60d2b03168c4",
      "name": "10Mbit",
      "description": "This policy limits the ports to 10Mbit max.",
      "shared": false,
      "rules": [
        {
          "max_kbps": 10000,
          "type": "bandwidth_limit",
          "id": "b1866696-032a-4228-857f-846075f63487",
          "max_burst_kbps": 0,
          "qos_policy_id": "46ebaec0-0570-43ac-82f6-60d2b03168c4"
        },
        {
          "dscp_mark": 20,
          "type": "dscp_marking",
          "id": "d9c021d5-5433-4d7c-8bfa-69cca486aac8",
          "qos_policy_id": "46ebaec0-0570-43ac-82f6-60d2b03168c4"
        }
      ]
    }
  ]
}
```

### POST

/v2.0/qos/policies

## Create QoS policy

Creates a QoS policy.

Creates a QoS policy by using the configuration that you define in the request object. A response object is returned. The object contains a unique ID.

If the caller is not an administrative user, this call returns the HTTP Forbidden (403) response code.

Users with an administrative role can create policies on behalf of other projects by specifying a project ID that is different than their own.

Normal response codes: 201

Error response codes: 401, 404, 409, 413

## Request

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
policy	body	object	A QoS policy object.

Name	In	Type	Description
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
name	body	string	Human-readable name of the resource.

### Request Example

```
{
  "policy": {
    "name": "10Mbit",
    "description": "This policy limits the ports to 10Mbit max.",
    "shared": false
  }
}
```

### Response Parameters

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
policy	body	object	A QoS policy object.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
id	body	string	The ID of the resource.
name	body	string	Human-readable name of the resource.

**GET**

/v2.0/qos/policies/{policy\_id}/dscp\_marking\_rules/{dscp\_rule\_id}

### Show DSCP marking rule details

Shows details for a DSCP marking rule for a QoS policy.

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
dscp_rule_id	path	string	The ID of the DSCP rule.
policy_id (Optional)	path	string	The UUID of the policy.

## Response Parameters

Name	In	Type	Description
dscp_marking_rule	body	object	A dscp_marking_rule object.
dscp_mark (Optional)	body	integer	The DSCP mark value.
id	body	string	The ID of the resource.
policy_id (Optional)	path	string	The UUID of the policy.

## Response Example

```
{
  "dscp_marking_rule": {
    "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c794",
    "dscp_mark": 26
  }
}
```

## PUT

/v2.0/qos/policies/{policy\_id}/dscp\_marking\_rules/{dscp\_rule\_id}

## Update DSCP marking rule

Updates a DSCP marking rule for a QoS policy.

If the request is valid, the service returns the Accepted (202) response code.

Normal response codes: 200

Error response codes: 400, 401, 413

## Request

Name	In	Type	Description
dscp_marking_rule	body	object	A dscp_marking_rule object.
dscp_mark (Optional)	body	integer	The DSCP mark value.
policy_id (Optional)	path	string	The UUID of the policy.
dscp_rule_id	path	string	The ID of the DSCP rule.

Name	In	Type	Description

### Request Example

```
{
  "dscp_marking_rule": {
    "dscp_mark": "16"
  }
}
```

### Response Parameters

Name	In	Type	Description
dscp_marking_rule	body	object	A dscp_marking_rule object.
dscp_mark (Optional)	body	integer	The DSCP mark value.
id	body	string	The ID of the resource.
policy_id (Optional)	path	string	The UUID of the policy.

### Response Example

```
{
  "dscp_marking_rule": {
    "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c794",
    "dscp_mark": 16
  }
}
```

### DELETE

/v2.0/qos/policies/{policy\_id}/dscp\_marking\_rules/{dscp\_rule\_id}

### Delete DSCP marking rule

Deletes a DSCP marking rule for a QoS policy.

Normal response codes: 204

Error response codes: 400, 401, 413

### Request

Name	In	Type	Description
policy_id (Optional)	path	string	The UUID of the policy.
dscp_rule_id	path	string	The ID of the DSCP rule.

### GET

/v2.0/qos/policies/{policy\_id}/dscp\_marking\_rules

### List DSCP marking rules for QoS policy

Lists all DSCP marking rules for a QoS policy.

The list may be empty.

Normal response codes: 200

Error response codes: 401

Request

Name	In	Type	Description
policy_id (Optional)	path	string	The UUID of the policy.

Response Parameters

Name	In	Type	Description
dscp_marking_rules	body	array	A list of dscp_marking_rule objects.
dscp_mark (Optional)	body	integer	The DSCP mark value.
id	body	string	The ID of the resource.
policy_id (Optional)	path	string	The UUID of the policy.

Response Example

```
{
  "dscp_marking_rules": [
    {
      "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c794",
      "dscp_mark": 26
    }
  ]
}
```

**POST**

/v2.0/qos/policies/{policy\_id}/dscp\_marking\_rules

### Create DSCP marking rule

Creates a DSCP marking rule for a QoS policy.

Normal response codes: 201

Error response codes: 401, 404, 409, 413

Request

Name	In	Type	Description
dscp_marking_rule	body	object	A dscp_marking_rule object.

Name	In	Type	Description
dscp_mark (Optional)	body	integer	The DSCP mark value.
policy_id (Optional)	path	string	The UUID of the policy.

### Request Example

```
{
  "dscp_marking_rule": {
    "dscp_mark": "26"
  }
}
```

### Response Parameters

Name	In	Type	Description
dscp_marking_rule	body	object	A dscp_marking_rule object.
dscp_mark (Optional)	body	integer	The DSCP mark value.
id	body	string	The ID of the resource.
policy_id (Optional)	path	string	The UUID of the policy.

**GET**

/v2.0/qos/policies/{policy\_id}

### Show QoS policy details

Shows details for a QoS policy.

Normal response codes: 200

Error response codes: 401

### Request

Name	In	Type	Description
policy_id (Optional)	path	string	The UUID of the policy.

### Response Parameters

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
policy_id (Optional)	path	string	The UUID of the policy.
tenant_id	body	string	The ID of the project.



Name	In	Type	Description
project_id	body	string	The ID of the project.
policy	body	object	A QoS policy object.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
rules (Optional)	body	array	A set of zero or more policy rules.
id	body	string	The ID of the resource.
name	body	string	Human-readable name of the resource.

## Response Example

```
{
  "policy": {
    "project_id": "8d4c70a21fed4aeba121a1a429ba0d04",
    "tenant_id": "8d4c70a21fed4aeba121a1a429ba0d04",
    "id": "46ebaec0-0570-43ac-82f6-60d2b03168c4",
    "name": "10Mbit",
    "description": "This policy limits the ports to 10Mbit max.",
    "shared": false,
    "rules": [
      {
        "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c793",
        "qos_policy_id": "46ebaec0-0570-43ac-82f6-60d2b03168c4",
        "max_kbps": 10000,
        "max_burst_kbps": 0,
        "type": "bandwidth_limit"
      },
      {
        "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c794",
        "qos_policy_id": "46ebaec0-0570-43ac-82f6-60d2b03168c4",
        "dscp_mark": 26,
        "type": "dscp_marking"
      }
    ]
  }
}
```

**PUT**

/v2.0/qos/policies/{policy\_id}

## Update QoS policy

Updates a QoS policy.

If the request is valid, the service returns the Accepted (202) response code.

Normal response codes: 200

Error response codes: 400, 401, 413

## Request

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
policy	body	object	A QoS policy object.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
name	body	string	Human-readable name of the resource.
policy_id (Optional)	path	string	The UUID of the policy.

## Request Example

```
{
  "policy": {
    "name": "10Mbit",
    "description": "This policy limits the ports to 10Mbit max.",
    "shared": false
  }
}
```

## Response Parameters

Name	In	Type	Description
description	body	string	A human-readable description for the resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
policy	body	object	A QoS policy object.
shared (Optional)	body	boolean	Admin-only. Indicates whether this network is shared across all projects.
id	body	string	The ID of the resource.
name	body	string	Human-readable name of the

Name	In	Type	Description
			resource.

## Response Example

```
{
  "policy": {
    "name": "10Mbit",
    "description": "This policy limits the ports to 10Mbit max.",
    "id": "46ebaec0-0570-43ac-82f6-60d2b03168c4",
    "project_id": "8d4c70a21fed4aeba121a1a429ba0d04",
    "tenant_id": "8d4c70a21fed4aeba121a1a429ba0d04",
    "shared": false
  }
}
```

## DELETE

/v2.0/qos/policies/{policy\_id}

## Delete QoS policy

Deletes a QoS policy.

Normal response codes: 204

Error response codes: 400, 401, 413

## Request

Name	In	Type	Description
policy_id (Optional)	path	string	The UUID of the policy.

## GET

/v2.0/qos/policies/{policy\_id}/bandwidth\_limit\_rules

## List bandwidth limit rules for QoS policy

Lists all bandwidth limit rules for a QoS policy.

The list might be empty.

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
policy_id (Optional)	path	string	The UUID of the policy.

## Response Parameters

Name	In	Type	Description
max_kbps (Optional)	body	integer	The maximum KBPS value. If you specify this value, must be greater than 0. Default is null.
bandwidth_limit_rules	body	array	A list of bandwidth limit rules associated with the QoS policy.
id	body	string	The ID of the resource.
max_burst_kbps (Optional)	body	integer	The burst over the maximum KBPS value. Default is 0
policy_id (Optional)	path	string	The UUID of the policy.

## Response Example

```
{
  "bandwidth_limit_rules": [
    {
      "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c793",
      "max_kbps": 10000,
      "max_burst_kbps": 0
    }
  ]
}
```

## POST

/v2.0/qos/policies/{policy\_id}/bandwidth\_limit\_rules

## Create bandwidth limit rule

Creates a bandwidth limit rule for a QoS policy.

Normal response codes: 201

Error response codes: 401, 404, 409, 413

## Request

Name	In	Type	Description
bandwidth_limit_rule	body	object	A bandwidth_limit_rule object.
max_kbps (Optional)	body	integer	The maximum KBPS value. If you specify this value, must be greater than 0. Default is null.
max_burst_kbps (Optional)	body	integer	The burst over the maximum KBPS value.

Name	In	Type	Description
			Default is 0
policy_id (Optional)	path	string	The UUID of the policy.

### Request Example

```
{
  "bandwidth_limit_rule": {
    "max_kbps": "10000"
  }
}
```

### Response Parameters

Name	In	Type	Description
bandwidth_limit_rule	body	object	A bandwidth_limit_rule object.
max_kbps (Optional)	body	integer	The maximum KBPS value. If you specify this value, must be greater than 0. Default is null.
id	body	string	The ID of the resource.
max_burst_kbps (Optional)	body	integer	The burst over the maximum KBPS value. Default is 0
policy_id (Optional)	path	string	The UUID of the policy.

# Load Balancer as a Service

## LBaaS 2.0 (STABLE)

The Load-Balancer-as-a-Service (LBaaS) version 2.0 extension pairs with the Networking 2.0 API to enable OpenStack projects to manage load balancers for their VMs. With this extension you can load-balance client traffic from one network to application services, such as VMs, on the same network.

Use this extension to create and manage load balancers, listeners, pools, members of a pool, and health monitors and view status of a resource.

### Load balancer statuses

Status	Description
ACTIVE	The resource is ready and active.
PENDING_CREATE	The resource is being created.

PENDING_UPDATE	The resource is being updated.
PENDING_DELETE	The resource is pending deletion.
INACTIVE	The resource is not active.
ERROR	An object within the service is not working. The <code>error_details</code> attribute provides an explanation for the error, its cause, and possibly a solution.

**GET**

/v2.0/lbaas/loadbalancers

## List load balancers

Lists all load balancers for the project.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

The list might be empty.

Normal response codes: 200

Error response codes: 401

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
loadbalancers	body	array	A list of load balancer objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the

Name	In	Type	Description
			resource.
flavor	body	string	The ID of the flavor.
id	body	string	The ID of the load balancer.
listeners	body	array	The associated listeners, if any.
name	body	string	Human-readable name of the resource.
operating_status	body	string	The operating status of the load balancer. This value is ONLINE or OFFLINE.
project_id	body	string	The ID of the project.
provider	body	string	Provider name of the load balancer service.
provisioning_status	body	string	The provisioning status of the load balancer. This value is ACTIVE, PENDING_CREATE or ERROR.
tenant_id	body	string	The ID of the project.
vip_address	body	string	The IP address of the VIP .
vip_subnet_id (Optional)	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address. This option is required if no vip_network_id is given.
pools	body	array	A list of pool objects.

## Response Example

```
{
  "loadbalancers": [
    {
      "description": "simple lb",
      "admin_state_up": true,
      "project_id": "1a3e005cf9ce40308c900bcb08e5320c",
      "tenant_id": "1a3e005cf9ce40308c900bcb08e5320c",
      "provisioning_status": "ACTIVE",
      "listeners": [],
      "vip_address": "10.0.0.2",
      "vip_subnet_id": "013d3059-87a4-45a5-91e9-d721068ae0b2",
      "id": "a9729389-6147-41a3-ab22-a24aed8692b2",
      "operating_status": "ONLINE",
      "name": "loadbalancer1",
      "flavor": "a7ae5d5a-d855-4f9a-b187-af66b53f4d04",
      "provider": "sample-provider",
      "pools": []
    }
  ]
}
```

**POST**  
/v2.0/lbaas/loadbalancers

## Create a load balancer

Creates a load balancer.

This operation provisions a new load balancer by using the configuration that you define in the request object. After the API validates the request and starts the provisioning process, the API returns a response object that contains a unique ID and the status of provisioning the load balancer.

In the response, the load balancer provisioning status is `ACTIVE`, `PENDING_CREATE`, or `ERROR`.

If the status is `PENDING_CREATE`, issue `GET /lbaas/loadbalancers/loadbalancer_id` to view the progress of the provisioning operation. When the load balancer status changes to `ACTIVE`, the load balancer is successfully provisioned and operational for traffic handling.

If the API cannot fulfill the request due to insufficient data or data that is not valid, the service returns the HTTP Bad Request (400) response code with information about the failure in the response body. Validation errors require that you correct the error and submit the request again.

You can configure all documented features of the load balancer at creation time by specifying the additional elements or attributes in the request.

Administrative users can specify a project ID that is different than their own to create load balancers for other projects.

### Example: Create a load balancer

- `project_id`. Admin only. Required to create a load balancer for another project.
- `vip_subnet_id`. The network on which to allocate the VIP address for the load balancer. A project can only create load balancer VIPs on networks that the policy authorizes, such as her own networks or shared or provider networks.

Some attributes receive default values if you omit them from the request:

- `admin_state_up`. Default is `true`.
- `name`. Default is an empty string.
- `description`. Default is an empty string.

If you own the subnet where you want to create the load balancer VIP, you can specify a `vip_address` attribute. If you omit the `vip_address` attribute from the payload, the LBaaS service allocates a VIP address from the subnet of the load balancer VIP.

An optional `flavor` attribute can be passed to enable dynamic selection of an appropriate provider if configured by the operator. The basic selection algorithm chooses the provider in the first service profile currently associated with flavor.

You can also specify the `provider` attribute when you create a load balancer. You can set this attribute to any service provider with a `LOADBALANCER` service type. Setting both a `flavor` and a `provider` will result in a conflict error.



Finally, `vip_network_id` can be used in place of `vip_subnet_id`. When this option is used, the VIP port is created on the given network using the default behavior. If assigned multiple fixed IPs, an IPv4 addresses is chosen as the VIP in preference to IPv6 addresses.

Normal response codes: 201

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
loadbalancer	body	object	A loadbalancer object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
flavor (Optional)	body	string	The ID of the flavor.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
provider (Optional)	body	string	Provider name of the load balancer service.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
vip_address	body	string	The IP address of the VIP .
vip_subnet_id (Optional)	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address. This option is required if no <code>vip_network_id</code> is given.

## Request Example

```
{
  "loadbalancer": {
    "name": "loadbalancer1",
    "description": "simple lb",
    "project_id": "b7c1a69e88bf4b21a8148f787aef2081",
    "tenant_id": "b7c1a69e88bf4b21a8148f787aef2081",
    "vip_subnet_id": "013d3059-87a4-45a5-91e9-d721068ae0b2",
    "vip_address": "10.0.0.4",
    "admin_state_up": true,
  }
}
```

```
"flavor": "a7ae5d5a-d855-4f9a-b187-af66b53f4d04"
```

```
}
```

## Response Parameters

Name	In	Type	Description
loadbalancer	body	object	A loadbalancer object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
flavor	body	string	The ID of the flavor.
id	body	string	The ID of the load balancer.
listeners	body	array	The associated listeners, if any.
name	body	string	Human-readable name of the resource.
operating_status	body	string	The operating status of the load balancer. This value is ONLINE or OFFLINE.
project_id	body	string	The ID of the project.
provider	body	string	Provider name of the load balancer service.
provisioning_status	body	string	The provisioning status of the load balancer. This value is ACTIVE, PENDING_CREATE or ERROR.
tenant_id	body	string	The ID of the project.
vip_address	body	string	The IP address of the VIP .
vip_network_id (Optional)	body	string	The ID of the network on which to allocate the virtual IP (VIP) address. This option is required if no vip_subnet_id is given.
vip_subnet_id (Optional)	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address. This option is required if no vip_network_id is given.
pools	body	array	A list of pool objects.

Name	In	Type	Description

## Response Example

```
{
  "loadbalancer": {
    "admin_state_up": true,
    "description": "simple lb",
    "id": "a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "listeners": [],
    "name": "loadbalancer1",
    "operating_status": "ONLINE",
    "provisioning_status": "ACTIVE",
    "project_id": "b7c1a69e88bf4b21a8148f787aef2081",
    "tenant_id": "b7c1a69e88bf4b21a8148f787aef2081",
    "vip_address": "10.0.0.4",
    "vip_subnet_id": "013d3059-87a4-45a5-91e9-d721068ae0b2",
    "flavor": "a7ae5d5a-d855-4f9a-b187-af66b53f4d04",
    "provider": "sample_provider",
    "pools": []
  }
}
```

## GET

/v2.0/lbaas/loadbalancers/{loadbalancer\_id}

## Show load balancer details

Shows details for a load balancer.

This operation returns a load balancer object, by ID. If you are not an administrative user and the load balancer object does not belong to your project, the service returns the HTTP Forbidden (403) response code.

Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
loadbalancer_id	path	string	The ID of the load balancer.

## Response Parameters

Name	In	Type	Description
loadbalancer	body	object	A loadbalancer object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description

Name	In	Type	Description
			for the resource.
id	body	string	The ID of the load balancer.
listeners	body	array	The associated listeners, if any.
name	body	string	Human-readable name of the resource.
operating_status	body	string	The operating status of the load balancer. This value is ONLINE or OFFLINE.
project_id	body	string	The ID of the project.
provisioning_status	body	string	The provisioning status of the load balancer. This value is ACTIVE, PENDING_CREATE or ERROR.
tenant_id	body	string	The ID of the project.
vip_address	body	string	The IP address of the VIP .
vip_subnet_id (Optional)	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address. This option is required if no vip_network_id is given.
pools	body	array	A list of pool objects.

## Response Example

```
{
  "loadbalancer": {
    "description": "simple lb",
    "admin_state_up": true,
    "project_id": "1a3e005cf9ce40308c900bcb08e5320c",
    "tenant_id": "1a3e005cf9ce40308c900bcb08e5320c",
    "provisioning_status": "ACTIVE",
    "listeners": [],
    "vip_address": "10.0.0.2",
    "vip_subnet_id": "013d3059-87a4-45a5-91e9-d721068ae0b2",
    "id": "a9729389-6147-41a3-ab22-a24aed8692b2",
    "operating_status": "ONLINE",
    "name": "loadbalancer1",
    "pools": []
  }
}
```

**PUT**

/v2.0/lbaas/loadbalancers/{loadbalancer\_id}

## Update load balancer

Updates a load balancer.

If the request is valid, the service returns the Accepted (202) response code. To confirm the update, check that the load balancer provisioning status is ACTIVE. If the status is PENDING\_UPDATE, use a GET operation to poll the load balancer object for changes.

This operation returns the updated load balancer object with the ACTIVE, PENDING\_UPDATE, or ERROR provisioning status.

Normal response codes: 202

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
loadbalancer_id	path	string	The ID of the load balancer.
loadbalancer	body	object	A loadbalancer object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.

## Request Example

```
{
  "loadbalancer": {
    "admin_state_up": false,
    "description": "simple lb2",
    "name": "loadbalancer2"
  }
}
```

## Response Parameters

Name	In	Type	Description
loadbalancer	body	object	A loadbalancer object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
flavor	body	string	The ID of the flavor.

Name	In	Type	Description
id	body	string	The ID of the load balancer.
listeners	body	array	The associated listeners, if any.
name	body	string	Human-readable name of the resource.
operating_status	body	string	The operating status of the load balancer. This value is ONLINE or OFFLINE.
project_id	body	string	The ID of the project.
provider	body	string	Provider name of the load balancer service.
provisioning_status	body	string	The provisioning status of the load balancer. This value is ACTIVE, PENDING_CREATE or ERROR.
tenant_id	body	string	The ID of the project.
vip_address	body	string	The IP address of the VIP .
vip_subnet_id (Optional)	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address. This option is required if no vip_network_id is given.
pools	body	array	A list of pool objects.

## Response Example

```
{
  "loadbalancer": {
    "admin_state_up": false,
    "description": "simple lb2",
    "id": "a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "listeners": [],
    "name": "loadbalancer2",
    "operating_status": "ONLINE",
    "provisioning_status": "PENDING_UPDATE",
    "project_id": "b7c1a69e88bf4b21a8148f787aef2081",
    "tenant_id": "b7c1a69e88bf4b21a8148f787aef2081",
    "vip_address": "10.0.0.4",
    "vip_subnet_id": "013d3059-87a4-45a5-91e9-d721068ae0b2",
    "flavor": "a7ae5d5a-d855-4f9a-b187-af66b53f4d04",
    "provider": "sample_provider",
    "pools": []
  }
}
```

**DELETE**

/v2.0/lbaas/loadbalancers/{loadbalancer\_id}

## Remove load balancer

Removes a load balancer and its associated configuration from the project.

The API immediately purges any and all configuration data. You cannot recover it.

Example: Delete a load balancer

Normal response codes: 204

Error response codes: 401, 404

### Request

Name	In	Type	Description
loadbalancer_id	path	string	The ID of the load balancer.

### Response

There is no body content for the response of a successful DELETE request.

**GET**

/v2.0/lbaas/loadbalancers/{loadbalancer\_id}/statuses

## Show load balancer status tree

Shows the status tree for a load balancer.

This operation returns a status tree for a load balancer object, by load balancer ID. If you are not an administrative user and the load balancer object does not belong to the project, the API returns the Forbidden (403) response code.

If the operation succeeds, the returned element is a status tree that contains the load balancer and all provisioning and operating statuses for its children.

Normal response codes: 200

Error response codes: 401, 403

### Request

Name	In	Type	Description
loadbalancer_id	path	string	The ID of the load balancer.

### Response Parameters

Name	In	Type	Description
statuses	body	object	The status tree of a load balancer object contains all provisioning and operating

Name	In	Type	Description
			statuses for its children.
loadbalancer	body	object	A loadbalancer object.
listeners	body	array	The associated listeners, if any.
pools	body	array	List of pools that are associated with the health monitor.
healthmonitor	body	object	The associated healthmonitor, if any.
id	body	string	The ID of the load balancer.
members	body	array	The list of members that belong to the pool.
operating_status	body	string	The operating status of the load balancer. This value is ONLINE or OFFLINE.
provisioning_status	body	string	The provisioning status of the load balancer. This value is ACTIVE, PENDING_CREATE or ERROR.

## Response Example

```
{
  "statuses": {
    "loadbalancer": {
      "name": "lb1",
      "listeners": [
        {
          "pools": [
            {
              "name": "pool1",
              "provisioning_status": "ACTIVE",
              "health_monitor": {
                "type": "HTTP",
                "id": "90f7c765-0bc9-47c4-8513-4cc0c264c8f8",
                "provisioning_status": "ACTIVE"
              }
            },
            {
              "members": [
                {
                  "address": "10.0.0.4",
                  "protocol_port": 80,
                  "id": "32723bee-2484-4de3-b6fc-c0b98d35fc84",
                  "operating_status": "ONLINE",
                  "provisioning_status": "ACTIVE"
                },
                {
                  "address": "10.0.0.3",
                  "protocol_port": 80,
                  "id": "173b8164-0c9a-43ec-ab33-4ae0e7a8f863",
                  "operating_status": "ONLINE",
                  "provisioning_status": "ACTIVE"
                }
              ]
            }
          ]
        }
      ]
    },
    "id": "ae6f93b8-a3f6-46cd-bb18-c2ab0308abf7",
    "operating_status": "ONLINE"
  }
}
```



```

    },
    "name": "listener1",
    "id": "c2a41fbe-b70a-4645-bb11-4d3c28f23a25",
    "operating_status": "ONLINE",
    "provisioning_status": "ACTIVE"
  },
  {
    "id": "a4c19566-6f81-4c96-ac11-33954a9825a2",
    "operating_status": "ONLINE",
    "provisioning_status": "ACTIVE"
  }
}

```

**GET**

/v2.0/lbaas/listeners

## List listeners

Lists all listeners.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

This operation lists all listeners that are associated with your project.

The list might be empty.

Example: List listeners

Normal response codes: 200

Error response codes: 401

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response Parameters

Name	In	Type	Description
listeners	body	array	A list of listeners objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).

Name	In	Type	Description
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
default_pool_id	body	string	The ID of default pool. Must have compatible protocol with listener.
default_tls_container_ref	body	string	A reference to a container of TLS secrets.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the listener.
loadbalancers	body	array	A list of load balancer objects.
name	body	string	Human-readable name of the resource.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
protocol_port	body	integer	The TCP or UDP port on which to listen.
sni_container_refs	body	array	A list of references to TLS secrets.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "listeners": [
    {
      "admin_state_up": true,
      "connection_limit": 100,
      "default_pool_id": null,
      "description": "",
      "id": "35cb8516-1173-4035-8dae-0dae3453f37f",
      "loadbalancers": [
        {
          "id": "a9729389-6147-41a3-ab22-a24aed8692b2"
        }
      ],
      "name": "",
      "protocol": "HTTP",
      "protocol_port": 80,
      "project_id": "3e4d8bec50a845fcb09e03a4375c691d",
      "tenant_id": "3e4d8bec50a845fcb09e03a4375c691d",
      "default_tls_container_ref":
"https://barbican.endpoint/containers/a36c20d0-18e9-42ce-88fd-82a35977ee8c",
      "sni_container_refs": [
        "https://barbican.endpoint/containers/b36c20d0-18e9-42ce-88fd-82a35977ee8d",
        "https://barbican.endpoint/containers/c36c20d0-18e9-42ce-88fd-82a35977ee8e"
      ]
    }
  ]
}
```

}

**POST**

/v2.0/lbaas/listeners

## Create listener

Creates a listener.

This operation provisions a new listener by using the configuration that you define in the request object. After the request is validated and the provisioning process begins, a response object is returned. The object contains a unique identifier.

At a minimum, you must specify these listener attributes:

- `project_id`. Admin only. Required to create a listener for another project.
- `loadbalancer_id`. The load balancer on which to provision this listener. A project can only create listeners on load balancers that the policy authorizes. For example, her own load balancers.
- `description`. The load balancer description.
- `protocol`. The protocol for which the front end listens. Must be HTTP, HTTPS, TCP, or TERMINATED\_HTTPS.

Some attributes receive default values if you omit them from the request:

- `protocol_port`. The port on which the front end listens. Must be an integer from 1 to 65535.
- `default_tls_container_ref`. The reference to a container that holds TLS secrets. If you also specify `sni_container_refs`, this container is the default. This parameter is required for the TERMINATED\_HTTPS protocol.
- `sni_container_refs`. A list of references to containers that hold TLS secrets for server name indication (SNI). This parameter is required for the TERMINATED\_HTTPS protocol.
- `admin_state_up`. Default is true.
- `name`. Default is an empty string.
- `description`. Default is an empty string.
- `connection_limit`. Default is -1, which indicates an infinite limit.

If the API cannot fulfill the request due to insufficient data or data that is not valid, the service returns the HTTP Bad Request (400) response code with information about the failure in the response body. Validation errors require that you correct the error and submit the request again.

You can configure all documented features of the listener at creation time by specifying the additional elements or attributes in the request.

Administrative users can specify a project ID that is different than their own to create listeners for other projects.

To update a listener, the load balancer to which to attach must have an ACTIVE provisioning status.

Normal response codes: 201

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
listener	body	object	A listener object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
connection_limit (Optional)	body	integer	The maximum number of connections permitted for this load balancer. Default is infinite.
default_pool_id (Optional)	body	string	The ID of default pool. Must have compatible protocol with listener.
default_tls_container_ref (Optional)	body	string	A reference to a container of TLS secrets.
description	body	string	A human-readable description for the resource.
loadbalancer_id	body	string	The ID of the load balancer.
name	body	string	Human-readable name of the resource.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
protocol_port (Optional)	body	integer	The TCP or UDP port on which to listen.
sni_container_refs (Optional)	body	array	A list of references to TLS secrets.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

## Request Example

```
{
  "listener": {
    "admin_state_up": true,
    "connection_limit": 100,
    "description": "listener one",
    "loadbalancer_id": "a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "name": "listener1",
    "protocol": "HTTP",
    "protocol_port": "80",
    "default_tls_container_ref": "https://barbican.endpoint/containers/a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "sni_container_refs": [
      "https://barbican.endpoint/containers/b36c20d0-18e9-42ce-88fd-
```

```

82a35977ee8d"/
      "https://barbican.endpoint/containers/c36c20d0-18e9-42ce-88fd-
82a35977ee8e"
    ]
  }
}

```

## Response Parameters

Name	In	Type	Description
listener	body	object	A listener object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
default_pool_id	body	string	The ID of default pool. Must have compatible protocol with listener.
default_tls_container_ref	body	string	A reference to a container of TLS secrets.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the listener.
loadbalancers	body	array	A list of load balancer objects.
name	body	string	Human-readable name of the resource.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
protocol_port	body	integer	The TCP or UDP port on which to listen.
sni_container_refs	body	array	A list of references to TLS secrets.
tenant_id	body	string	The ID of the project.

## Response Example

```

{
  "listener": {
    "admin_state_up": true,
    "connection_limit": 100,
    "default_pool_id": null,
    "description": "listener one",
    "id": "39de4d56-d663-46e5-85a1-5b9d5fa17829",
    "loadbalancers": [
      {
        "id": "a36c20d0-18e9-42ce-88fd-82a35977ee8c"
      }
    ]
  }
}

```

```

    },
    "name": "listener1",
    "protocol": "HTTP",
    "protocol_port": 80,
    "project_id": "1a3e005cf9ce40308c900bcb08e5320c",
    "tenant_id": "1a3e005cf9ce40308c900bcb08e5320c",
    "default_tls_container_ref": "https://barbican.endpoint/containers/a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "sni_container_refs": [
        "https://barbican.endpoint/containers/b36c20d0-18e9-42ce-88fd-82a35977ee8d",
        "https://barbican.endpoint/containers/c36c20d0-18e9-42ce-88fd-82a35977ee8e"
    ]
}
}
}

```

**GET**

/v2.0/lbaas/listeners/{listener\_id}

## Show listener details

Shows details for a listener.

This operation returns a listener object, by ID. If you are not an administrative user and the listener object does not belong to your account, the API returns the HTTP Forbidden (403) response code.

Example: Show listener details

Normal response codes: 200

Error response codes: 401, 403

### Request

Name	In	Type	Description
listener_id	path	string	The ID of the listener.

### Response Parameters

Name	In	Type	Description
listener	body	object	A listener object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
id	body	string	The ID of the listener.
default_pool_id	body	string	The ID of default pool. Must have

Name	In	Type	Description
			compatible protocol with listener.
default_tls_container_ref	body	string	A reference to a container of TLS secrets.
description	body	string	A human-readable description for the resource.
loadbalancers	body	array	A list of load balancer objects.
name	body	string	Human-readable name of the resource.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
protocol_port	body	integer	The TCP or UDP port on which to listen.
sni_container_refs	body	array	A list of references to TLS secrets.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "listener": {
    "admin_state_up": true,
    "connection_limit": 100,
    "default_pool_id": null,
    "description": "",
    "id": "35cb8516-1173-4035-8dae-0dae3453f37f",
    "loadbalancers": [
      {
        "id": "a9729389-6147-41a3-ab22-a24aed8692b2"
      }
    ],
    "name": "",
    "protocol": "HTTP",
    "protocol_port": 80,
    "project_id": "3e4d8bec50a845fcb09e03a4375c691d",
    "tenant_id": "3e4d8bec50a845fcb09e03a4375c691d",
    "default_tls_container_ref": "https://barbican.endpoint/containers/a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "sni_container_refs": [
      "https://barbican.endpoint/containers/b36c20d0-18e9-42ce-88fd-82a35977ee8d",
      "https://barbican.endpoint/containers/c36c20d0-18e9-42ce-88fd-82a35977ee8e"
    ]
  }
}
```

**PUT**

/v2.0/lbaas/listeners/{listener\_id}

## Update listener

Updates a listener.

This operation updates the attributes of a listener. Upon successful validation of the request, the service returns the HTTP Accepted (202) response code.

Note: You cannot update the `listener_id`, `project_id`, `loadbalancer_id`, `loadbalancers`, `default_pool_id`, `protocol`, and `protocol_port` attributes. Attempting to update an immutable attribute results in the HTTP Immutable (422) response code.

Note: You cannot update a listener if the load balancer to which the listener is attached does not have an ACTIVE provisioning status.

Normal response codes: 202

Error response codes: 401, 404, 422

### Request

Name	In	Type	Description
<code>listener_id</code>	path	string	The ID of the listener.
<code>listener</code>	body	object	A listener object.
<code>admin_state_up</code> (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
<code>connection_limit</code> (Optional)	body	integer	The maximum number of connections permitted for this load balancer. Default is infinite.
<code>default_tls_container_ref</code> (Optional)	body	string	A reference to a container of TLS secrets.
<code>description</code>	body	string	A human-readable description for the resource.
<code>name</code>	body	string	Human-readable name of the resource.
<code>sni_container_refs</code> (Optional)	body	array	A list of references to TLS secrets.

### Request Example

```
{
  "listener": {
    "admin_state_up": false,
    "connection_limit": 200,
    "description": "listener two",
    "name": "listener2",
    "default_tls_container_ref": "https://barbican.endpoint/containers/a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "sni_container_refs": [
      "https://barbican.endpoint/containers/b36c20d0-18e9-42ce-88fd-
```



```

82a35977ee8d"/
      "https://barbican.endpoint/containers/c36c20d0-18e9-42ce-88fd-
82a35977ee8e"
    }
  }
}

```

## Response Parameters

Name	In	Type	Description
listener	body	object	A listener object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
default_pool_id	body	string	The ID of default pool. Must have compatible protocol with listener.
default_tls_container_ref	body	string	A reference to a container of TLS secrets.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the listener.
loadbalancers	body	array	A list of load balancer objects.
name	body	string	Human-readable name of the resource.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
protocol_port	body	integer	The TCP or UDP port on which to listen.
sni_container_refs	body	array	A list of references to TLS secrets.
tenant_id	body	string	The ID of the project.

## Response Example

```

{
  "listener": {
    "admin_state_up": false,
    "connection_limit": 200,
    "default_pool_id": null,
    "description": "listener two",
    "id": "39de4d56-d663-46e5-85a1-5b9d5fa17829",
    "loadbalancers": [
      {
        "id": "a36c20d0-18e9-42ce-88fd-82a35977ee8c"
      }
    ]
  }
}

```

```

    },
    "name": "listener2",
    "protocol": "HTTP",
    "protocol_port": 80,
    "project_id": "1a3e005cf9ce40308c900bcb08e5320c",
    "tenant_id": "1a3e005cf9ce40308c900bcb08e5320c",
    "default_tls_container_ref": "https://barbican.endpoint/containers/a36c20d0-18e9-42ce-88fd-82a35977ee8c",
    "sni_container_refs": [
        "https://barbican.endpoint/containers/b36c20d0-18e9-42ce-88fd-82a35977ee8d",
        "https://barbican.endpoint/containers/c36c20d0-18e9-42ce-88fd-82a35977ee8e"
    ]
}
}
}

```

## DELETE

/v2.0/lbaas/listeners/{listener\_id}

### Remove listener

Removes a listener.

This operation removes a listener and its associated configuration from the project. The API immediately purges any and all configuration data. You cannot recover it.

You cannot delete a listener if the load balancer to which it is attached does not have an ACTIVE provisioning status.

Example: Delete a listener

Normal response codes: 204

Error response codes: 401, 404, 409

### Request

Name	In	Type	Description
listener_id	path	string	The ID of the listener.

### Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/lbaas/pools

### List pools

Lists all pools that are associated with your project.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

The list might be empty.

Example: List pools

Normal response codes: 200

Error response codes: 401

#### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using fields parameter, the API returns only the requested set of attributes. fields parameter can be specified multiple times. For example, if you specify fields=id&fields=name in the request URL, only id and name attributes will be returned.

#### Response Parameters

Name	In	Type	Description
pools	body	array	A list of pool objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_algorithm	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the

Name	In	Type	Description
			resource.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
provider	body	string	Provider name of the load balancer service.
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{
  "pools": [
    {
      "status": "ACTIVE",
      "lb_algorithm": "ROUND_ROBIN",
      "protocol": "HTTP",
      "description": "",
      "health_monitors": [
        "b7633ade-24dc-4d72-8475-06aa22be5412"
      ],
      "members": [
        "cf024846-7516-4e3a-b0fb-6590322c836f"
      ],
      "status_description": null,
      "id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
      "vip_id": "388c739a-6a57-4e74-bc7b-a5cd60248bba",
      "name": "pool1",
      "admin_state_up": true,
      "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
      "project_id": "eabfefa3fd1740a88a47ad98e132d238",
      "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    }
  ]
}
```

```

    "health_monitors_status": [
      {
        "monitor_id": "b7633ade-24dc-4d72-8475-06aa22be5412",
        "status": "ACTIVE",
        "status_description": null
      }
    ],
    "provider": "haproxy"
  ]
}

```

**POST**

/v2.0/lbaas/pools

## Create pool

Creates a pool.

This operation provisions a pool by using the configuration that you define in the request object. After the API validates the request and starts the provisioning process, the API returns a response object, which contains a unique ID.

At a minimum, you must specify these pool attributes:

- `project_id`. Admin only. Required to create a pool for another project.
- `protocol`. The protocol for which this pool and its members listen. A valid value is TCP, HTTP, or HTTPS.
- `lb_algorithm`. The load-balancer algorithm, such as ROUND\_ROBIN, LEAST\_CONNECTIONS, and SOURCE\_IP, that distributes traffic to the pool members. The load-balancer provider must support this algorithm.
- `listener_id`. The ID of the listener in which this pool becomes the default pool. Each listener has only one default pool.

Some attributes receive default values if you omit them from the request:

- `admin_state_up`. Default is true.
- `name`. Default is an empty string.
- `description`. Default is an empty string.
- `session_persistence`. Default is an empty dictionary.

If the API cannot fulfill the request due to insufficient data or data that is not valid, the service returns the HTTP Bad Request (400) response code with information about the failure in the response body. Validation errors require that you correct the error and submit the request again.

Users can configure all documented features at creation time by providing the additional elements or attributes in the request.

Administrative users can specify a project ID that is different than their own to create pools for other projects.

To update a pool, the load balancer to which to attach must have an ACTIVE provisioning status.

Normal response codes: 201

## Error response codes: 400, 401

### Request

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
lb_algorithm	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
listener_id	body	string	The ID of the listener.
name	body	string	Human-readable name of the resource.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
protocol (Optional)	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

### Request Example

```
{  
  "pool": {  
    "admin_state_up": true,  
    "description": "simple pool",  
  }  
}
```

```

    "lb_algorithm": "ROUND_ROBIN",
    "name": "my-pool",
    "protocol": "HTTP",
    "subnet_id": "e301aed0-d9e7-498a-977c-1bbfaf14ed5d",
    "listener_id": "39de4d56-d663-46e5-85a1-5b9d5fa17829"
  }
}

```

## Response Parameters

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_algorithm	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the resource.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
provider	body	string	Provider name of the load balancer service.
status	body	string	The status of the pool. Indicates whether the pool is operational.

Name	In	Type	Description
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{
  "pool": {
    "status": "PENDING_CREATE",
    "lb_algorithm": "ROUND_ROBIN",
    "protocol": "HTTP",
    "description": "simple pool",
    "health_monitors": [],
    "members": [],
    "status_description": null,
    "id": "af95e0ce-8a26-4f29-9524-db41e7769c73",
    "vip_id": null,
    "name": "my-pool",
    "admin_state_up": true,
    "subnet_id": "e301aed0-d9e7-498a-977c-1bbfaf14ed5d",
    "project_id": "eabfefaf3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefaf3fd1740a88a47ad98e132d238",
    "health_monitors_status": [],
    "provider": "haproxy"
  }
}
```

**GET**

/v2.0/lbaas/pools/{pool\_id}

## Show pool details

Shows details for a pool.

This operation shows details for a pool, by ID. If you are not an administrative user and the pool object does not belong to your project, the call returns the HTTP Forbidden (403) response code.

If this operation succeeds, it returns a pool element.

Example: Show pool details

Normal response codes: 200

Error response codes: 403



## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.

## Response Parameters

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
lb_algorithm	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
id	body	string	The ID of the pool.
name	body	string	Human-readable name of the resource.
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
provider	body	string	Provider name of the load

Name	In	Type	Description
			balancer service.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{
  "pool": {
    "status": "ACTIVE",
    "lb_algorithm": "ROUND_ROBIN",
    "protocol": "HTTP",
    "description": "",
    "health_monitors": [
      "b7633ade-24dc-4d72-8475-06aa22be5412"
    ],
    "members": [
      "cf024846-7516-4e3a-b0fb-6590322c836f"
    ],
    "status_description": null,
    "id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "vip_id": "388c739a-6a57-4e74-bc7b-a5cd60248bba",
    "name": "pool1",
    "admin_state_up": true,
    "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "health_monitors_status": [
      {
        "monitor_id": "b7633ade-24dc-4d72-8475-06aa22be5412",
        "status": "ACTIVE",
        "status_description": null
      }
    ],
    "provider": "haproxy"
  }
}
```

**PUT**

/v2.0/lbaas/pools/{pool\_id}

## Update pool

Updates a pool.

This operation updates the attributes of a pool. Upon successful validation of the request, the service returns the HTTP Accepted (202) response code.

Note: You cannot update the pool ID, project\_id, listener\_id, listeners, health\_monitor\_id, protocol, and members immutable attributes. If you try to update any of these attributes, the service returns the HTTP Immutable (422) response code .

Note: You cannot update a pool if the load balancer to which it is attached does not have an ACTIVE provisioning status.

Normal response codes: 202

Error response codes: 401, 404, 422

### Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
pool	body	object	A pool object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
lb_algorithm	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
name	body	string	Human-readable name of the resource.

### Request Example

```
{
  "pool": {
    "name": "SuperPool"
  }
}
```

### Response Parameters

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.

Name	In	Type	Description
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_algorithm	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the resource.
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
project_id	body	string	The ID of the project.
protocol	body	string	The IP protocol. Valid value is icmp, tcp, udp, or null. No default.
provider	body	string	Provider name of the load balancer service.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{
  "pool": {
    "status": "PENDING_UPDATE",
    "lb_algorithm": "ROUND_ROBIN",
```

```

    "protocol": "HTTP",
    "description": "",
    "health_monitors": [
        "b7633ade-24dc-4d72-8475-06aa22be5412"
    ],
    "members": [
        "cf024846-7516-4e3a-b0fb-6590322c836f"
    ],
    "status_description": null,
    "id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "vip_id": "388c739a-6a57-4e74-bc7b-a5cd60248bba",
    "name": "SuperPool",
    "admin_state_up": true,
    "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "health_monitors_status": [
        {
            "monitor_id": "b7633ade-24dc-4d72-8475-06aa22be5412",
            "status": "ACTIVE",
            "status_description": null
        }
    ],
    "provider": "haproxy"
}

```

## DELETE

/v2.0/lbaas/pools/{pool\_id}

## Remove pool

Removes a pool.

This operation removes a pool and its associated configuration from the project. The API immediately purges any and all configuration data. You cannot recover it.

You cannot delete a pool if the load balancer to which it is attached does not have an ACTIVE provisioning status.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/lbaas/pools/{pool\_id}/members

## List pool members

Lists members of a pool.

Lists all members that are associated with a pool that is associated with your project. The list of members includes only members that belong to the pool object identified by pool\_id.

The list might be empty.

Example: List pool members

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.

## Response Parameters

Name	In	Type	Description
members	body	array	The list of members that belong to the pool.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
id	body	string	The ID of the member.
project_id	body	string	The ID of the project.
protocol_port	body	integer	The TCP or UDP port on which to listen.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "members": [
    {
      "address": "10.0.0.8",
      "admin_state_up": true,
```

```

      "id": "9a7aff27-fd41-4ec1-ba4c-3eb92c629313",
      "protocol_port": 80,
      "subnet_id": "013d3059-87a4-45a5-91e9-d721068ae0b2",
      "project_id": "1a3e005cf9ce40308c900bcb08e5320c",
      "tenant_id": "1a3e005cf9ce40308c900bcb08e5320c",
      "weight": 1
    }
  ]
}

```

**POST**

/v2.0/lbaas/pools/{pool\_id}/members

## Add member to pool

Adds a member to a pool.

This operation provisions a member and adds it to a pool by using the configuration that you define in the request object. After the API validates the request and starts the provisioning process, it returns a response object, which contains a unique ID.

At a minimum, you must specify these pool attributes:

- `project_id`. Admin only. Required to create a pool for another project.
- `address`. The IP address of the member to receive traffic from the load balancer.
- `protocol_port`. The port on which the member listens for traffic.

Some attributes receive default values if you omit them from the request:

- `admin_state_up`. Default is `true`.
- `weight`. Default is `1`.

If you omit the `subnet_id` parameter, LBaaS uses the `vip_subnet_id` parameter value for the subnet UUID.

If the request fails due to incorrect data, the service returns the HTTP Bad Request (400) response code with information about the failure in the response body. Validation errors require that you correct the error and submit the request again.

To configure all documented member features at creation time, specify additional elements or attributes in the request.

Administrative users can specify a project ID that is different than their own to create members for other projects.

To update a member, the load balancer must have an ACTIVE provisioning status.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
<code>pool_id</code>	path	string	The ID for the pool.

Name	In	Type	Description
address	body	string	The IP address of the member.
member	body	object	A member object.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
protocol_port	body	integer	The TCP or UDP port on which to listen.
subnet_id (Optional)	body	string	If you omit this parameter, LBaaS uses the vip_subnet_id parameter value for the subnet UUID.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

## Request Example

```
{
  "member": {
    "address": "10.0.0.22",
    "admin_state_up": true,
    "protocol_port": "90",
    "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "weight": "1"
  }
}
```

## Response Parameters

Name	In	Type	Description
member	body	object	A member object.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
id	body	string	The ID of the member.
project_id	body	string	The ID of the project.



Name	In	Type	Description
protocol_port	body	integer	The TCP or UDP port on which to listen.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load-balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "member": {
    "admin_state_up": true,
    "weight": 1,
    "address": "10.0.1.22",
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "protocol_port": 90,
    "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
    "subnet_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
  }
}
```

## GET

/v2.0/lbaas/pools/{pool\_id}/members/{member\_id}

## Show pool member details

Shows details for a pool member.

This operation returns a member object identified by member\_id that belongs to a pool object identified by pool\_id. If you are not an administrative user and the pool or member object does not belong to your project, the service returns the HTTP Forbidden (403) response code.

If this operation succeeds, it returns a pool element.

Example: Show pool member details

Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
member_id	path	string	The ID for the member.

## Response Parameters

Name	In	Type	Description
member	body	object	A member object.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
id	body	string	The ID of the member.
project_id	body	string	The ID of the project.
protocol_port	body	integer	The TCP or UDP port on which to listen.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "member": {
    "admin_state_up": true,
    "weight": 1,
    "address": "10.0.1.22",
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "protocol_port": 90,
    "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
    "subnet_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
  }
}
```

**PUT**

/v2.0/lbaas/pools/{pool\_id}/members/{member\_id}

## Update pool member

Updates attributes for a pool member.

Upon successful validation of the request, the service returns the HTTP OK (200) response code.

Note: You cannot update the member ID, project\_id, address, protocol\_port, and subnet\_id attributes. If you attempt to update any of these attributes, the service returns the HTTP Immutable (422) response code.

Note: You cannot update a member if the attached load balancer does not have an ACTIVE provisioning status.

Normal response codes: 200

Error response codes: 401, 404, 422

### Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
member_id	path	string	The ID for the member.
member	body	object	A member object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
weight (Optional)	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256. The default is 1.

### Request Example

```
{
  "member": {
    "weight": 5
  }
}
```

### Response Parameters

Name	In	Type	Description
member	body	object	A member object.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource,

Name	In	Type	Description
			which is up (true) or down (false).
id	body	string	The ID of the member.
project_id	body	string	The ID of the project.
protocol_port	body	integer	The TCP or UDP port on which to listen.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "member": {
    "admin_state_up": true,
    "weight": 5,
    "address": "10.0.1.22",
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "protocol_port": 90,
    "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
    "subnet_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
  }
}
```

## DELETE

/v2.0/lbaas/pools/{pool\_id}/members/{member\_id}

## Remove member from pool

Removes a member from a pool and its associated configuration from the project.

The API immediately purges any and all configuration data. You cannot recover it.

You cannot delete a member if the attached load balancer does not have an ACTIVE provisioning status.

Example: Remove a member from a pool

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
member_id	path	string	The ID for the member.

## Response

There is no body content for the response of a successful DELETE request.

### GET

/v2.0/lbaas/healthmonitors

## List health monitors

Lists health monitors.

This operation lists all health monitors that are associated with your project.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

This operation returns a list, which might be empty.

Normal response codes: 200

Error response codes: 401

## Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response Parameters

Name	In	Type	Description
health_monitors	body	array	A list of <code>health_monitor</code> objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up ( <code>true</code> ) or down ( <code>false</code> ).
delay	body	integer	The time, in seconds, between sending

Name	In	Type	Description
			probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"> <li>• A single value, such as 200</li> <li>• A list, such as 200, 202</li> <li>• A range, such as 200-204</li> </ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
id	body	string	The ID of the associated health monitor.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pools	body	array	List of pools that are associated with the health monitor.
project_id	body	string	The ID of the project.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitors": [
    {
      "admin_state_up": true,
      "project_id": "eabfefa3fd1740a88a47ad98e132d238",
      "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
      "delay": 1,
      "expected_codes": "200,201,202",
      "max_retries": 5,
      "http_method": "GET",
      "timeout": 1,
      "pools": [
        {
```

```

        "status": "ACTIVE",
        "status_description": null,
        "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
    },
    "url_path": "/index.html",
    "type": "HTTP",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412"
}
]
}

```

**POST**

/v2.0/lbaas/health\_monitors

## Create health monitor

Creates a health monitor.

This operation provisions a health monitor by using the configuration that you define in the request object. After the API validates the request and start the provisioning process, it returns a response object. The object contains a unique identifier.

At a minimum, you must specify these health monitor attributes:

- `project_id`. Admin only. Required to create a health monitor for another project.
- `type`. The type of health monitor. A valid value is TCP, HTTP, or HTTPS.
- `delay`. The interval, in seconds, between health checks.
- `timeout`. The time, in seconds, after which a health check times out.
- `max_retries`. Number of failed health checks before marked as OFFLINE.
- `pool_id`. The pool to monitor.

Some attributes receive default values if you omit them from the request, and are only useful when you specify a health monitor type of HTTP(S):

- `http_method`. Default is GET.
- `url_path`. Default is /.
- `expected_codes`. The expected HTTP status codes to get from a successful health check. Default is 200.
- `admin_state_up`. Default is true.

If the API cannot fulfill the request due to insufficient data or data that is not valid, it returns the Bad Request (400) response code with information about the nature of the failure in the response body. Failures in the validation process are non-recoverable and require that you correct the cause of the failure and submit the request again.

You can configure all documented features of the health monitor at creation time by specifying the additional elements or attributes in the request.

Administrative users can specify a project ID that is different than their own to create health monitors for other projects.

To update a health monitor, the load balancer to which to attach must have an ACTIVE provisioning status.

Normal response codes: 201

Error response codes: 401, 404

## Request

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes (Optional)	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pool_id	path	string	The ID for the pool.
project_id	body	string	The ID of the project.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.



Name	In	Type	Description
url_path (Optional)	body	string	The HTTP path of the request sent by the monitor to test the health of a member. A valid value is a string that begins with a forward slash (/).

## Request Example

```
{
  "health_monitor": {
    "pool_id": "74aa2010-a59f-4d35-a436-60a6da882819",
    "admin_state_up": true,
    "delay": "1",
    "expected_codes": "200,201,202",
    "http_method": "GET",
    "max_retries": 5,
    "timeout": 1,
    "type": "HTTP",
    "url_path": "/index.html"
  }
}
```

## Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"> <li>A single value, such as 200</li> <li>A list, such as 200, 202</li> <li>A range, such as 200-204</li> </ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
id	body	string	The ID of the associated health monitor.
max_retries	body	integer	The number of allowed

Name	In	Type	Description
			connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pools	body	array	List of pools that are associated with the health monitor.
project_id	body	string	The ID of the project.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitor": {
    "admin_state_up": true,
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "delay": 1,
    "expected_codes": "200,201,202",
    "max_retries": 5,
    "http_method": "GET",
    "timeout": 1,
    "pools": [
      {
        "status": "ACTIVE",
        "status_description": null,
        "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
      }
    ],
    "url_path": "/index.html",
    "type": "HTTP",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412"
  }
}
```

**GET**

/v2.0/lbaas/health\_monitors/{health\_monitor\_id}

## Show health monitor details

Shows details for a health monitor.

This operation returns a health monitor object, by health monitor ID. If you are not an administrative user and the health monitor object does not belong to your project, the service returns the HTTP Forbidden (403) response code.

Example: Show health monitor details

Normal response codes: 200

Error response codes: 401, 403

#### Request

Name	In	Type	Description
health_monitor_id	path	string	The ID for the health monitor.

#### Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
id	body	string	The ID of the associated

Name	In	Type	Description
			health monitor.
pools	body	array	List of pools that are associated with the health monitor.
project_id	body	string	The ID of the project.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitor": {
    "admin_state_up": true,
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "delay": 1,
    "expected_codes": "200,201,202",
    "max_retries": 5,
    "http_method": "GET",
    "timeout": 1,
    "pools": [
      {
        "status": "ACTIVE",
        "status_description": null,
        "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
      }
    ],
    "url_path": "/index.html",
    "type": "HTTP",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412"
  }
}
```

**PUT**

/v2.0/lbaas/health\_monitors/{health\_monitor\_id}

## Update health monitor

Updates a health monitor.

Upon successful validation of the request, the service returns the HTTP Accepted (202) response code.

Note: The health monitor ID, `project_id`, `pool_id`, and `type` are immutable attributes and cannot be updated. If you specify an unsupported attribute, the service returns the HTTP Immutable (422) response code.

Normal response codes: 202

Error response codes: 401, 404, 422

### Request

Name	In	Type	Description
health_monitor_id	path	string	The ID for the health monitor.
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes (Optional)	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
url_path (Optional)	body	string	The HTTP path of the request

Name	In	Type	Description
			sent by the monitor to test the health of a member. A valid value is a string that begins with a forward slash (/).

## Request Example

```
{
  "health_monitor": {
    "admin_state_up": false,
    "delay": "2",
    "expected_codes": "200",
    "http_method": "POST",
    "max_retries": 2,
    "timeout": 2,
    "url_path": "/page.html"
  }
}
```

## Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"> <li>• A single value, such as 200</li> <li>• A list, such as 200, 202</li> <li>• A range, such as 200-204</li> </ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.

Name	In	Type	Description
id	body	string	The ID of the associated health monitor.
pools	body	array	List of pools that are associated with the health monitor.
project_id	body	string	The ID of the project.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitor": {
    "admin_state_up": false,
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "delay": 2,
    "expected_codes": "200",
    "max_retries": 2,
    "http_method": "POST",
    "timeout": 2,
    "pools": [
      {
        "status": "ACTIVE",
        "status_description": null,
        "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332"
      }
    ],
    "url_path": "/page.html",
    "type": "HTTP",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412"
  }
}
```

## DELETE

/v2.0/lbaas/health\_monitors/{health\_monitor\_id}

## Remove health monitor

Removes a health monitor and its associated configuration from the project.

The API immediately purges any and all configuration data. You cannot recover it.

You cannot delete a health monitor if the attached load balancer does not have an ACTIVE provisioning status.

Example: Delete a health monitor

Normal response codes: 204

Error response codes: 401, 404

Request

Name	In	Type	Description
health_monitor_id	path	string	The ID for the health monitor.

Response

There is no body content for the response of a successful DELETE request.

## LBaaS 1.0 (DEPRECATED)

The Load-Balancer-as-a-Service (LBaaS) v1.0 extension pairs with the Networking v2.0 API to enable OpenStack projects to manage load balancers for their VMs. With this extension, you can load-balance client traffic from one network to application services, such as VMs, on the same network.

Use this extension to create and manage virtual IP addresses (VIPs), pools, members of a pool, health monitors, and view status of a resource.

### Load balancer statuses

Status	Description
ACTIVE	The resource is ready and active.
PENDING_CREATE	The resource is being created.
PENDING_UPDATE	The resource is being updated.
PENDING_DELETE	The resource is pending deletion.
INACTIVE	The resource is not active.
ERROR	An object within the service is not working. The <code>error_details</code> attribute provides an explanation for the error, its cause, and possibly a solution.



GET

/v2.0/lb/pools

## List pools

Lists pools.

Normal response codes: 200

Error response codes: 401

Request

Response Parameters

Name	In	Type	Description
pools	body	array	A list of pool objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_statuses	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_method	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the resource.
protocol	body	string	The protocol of the pool, which is TCP, HTTP, or HTTPS.
provider	body	string	The provider name of the load balancer pool.

Name	In	Type	Description
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{
  "pools": [
    {
      "admin_state_up": true,
      "description": "",
      "health_monitors": [
        "b7633ade-24dc-4d72-8475-06aa22be5412"
      ],
      "health_monitors_status": [
        {
          "monitor_id": "b7633ade-24dc-4d72-8475-06aa22be5412",
          "status": "ACTIVE",
          "status_description": null
        }
      ],
      "id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
      "lb_method": "ROUND_ROBIN",
      "members": [
        "cf024846-7516-4e3a-b0fb-6590322c836f"
      ],
      "name": "pool1",
      "protocol": "HTTP",
      "provider": "haproxy",
      "status": "ACTIVE",
      "status_description": null,
      "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
      "tenant_id": "eabfef3fd1740a88a47ad98e132d238",
      "vip_id": "388c739a-6a57-4e74-bc7b-a5cd60248bba"
    }
  ]
}
```

**POST**

/v2.0/lb/pools

## Create a load balancer pool

Creates a load balancer pool.

Normal response codes: 201

Error response codes: 400, 401

## Request

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
health_monitors (Optional)	body	array	List of health monitors to be associated with the pool. The default is an empty list.
lb_method	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
protocol	body	string	The protocol of the pool, which is TCP, HTTP, or HTTPS.
provider (Optional)	body	string	The provider name of the load balancer pool.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

## Request Example

```
{
  "pool": {
    "admin_state_up": true,
    "description": "simple pool",
    "lb_method": "ROUND_ROBIN",
    "name": "my-pool",
    "protocol": "HTTP",
    "subnet_id": "e301aed0-d9e7-498a-977c-1bbfaf14ed5d"
  }
}
```

}

## Response Parameters

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_method	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the resource.
protocol	body	string	The protocol of the pool, which is TCP, HTTP, or HTTPS.
provider	body	string	The provider name of the load balancer pool.
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{  
  "pool": {  
    "admin_state_up": true,  
    "description": "simple pool",  
  }  
}
```

```

    "health_monitors": [],
    "health_monitors_status": [],
    "id": "af95e0ce-8a26-4f29-9524-db41e7769c73",
    "lb_method": "ROUND_ROBIN",
    "members": [],
    "name": "my-pool",
    "protocol": "HTTP",
    "provider": "haproxy",
    "status": "PENDING_CREATE",
    "status_description": null,
    "subnet_id": "e301aed0-d9e7-498a-977c-1bbfaf14ed5d",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "vip_id": null
  }
}

```

**GET**

/v2.0/lb/pools/{pool\_id}

## Show pool details

Shows details for a pool.

Normal response codes: 200

Error response codes: 401, 404

Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.

Response Parameters

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_method	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP

Name	In	Type	Description
			(SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the resource.
protocol	body	string	The protocol of the pool, which is TCP, HTTP, or HTTPS.
provider	body	string	The provider name of the load balancer pool.
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP) address.

## Response Example

```
{
  "pool": {
    "admin_state_up": true,
    "description": "",
    "health_monitors": [
      "b7633ade-24dc-4d72-8475-06aa22be5412"
    ],
    "health_monitors_status": [
      {
        "monitor_id": "b7633ade-24dc-4d72-8475-06aa22be5412",
        "status": "ACTIVE",
        "status_description": null
      }
    ],
    "id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "lb_method": "ROUND_ROBIN",
    "members": [
      "cf024846-7516-4e3a-b0fb-6590322c836f"
    ],
    "name": "pool1",
    "protocol": "HTTP",
    "provider": "haproxy",
    "status": "ACTIVE",
    "status_description": null,
    "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
  }
}
```

```

    "vip_id": "388c739a-6a57-4e74-bc7b-a5cd60248bba"
  }
}

```

## PUT

/v2.0/lb/pools/{pool\_id}

## Update pool

Updates a load balancer pool.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
pool	body	object	A pool object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
health_monitors (Optional)	body	array	List of health monitors to be associated with the pool. The default is an empty list.
lb_method (Optional)	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.

## Request Example

```

{
  "pool": {
    "name": "SuperPool"
  }
}

```

## Response Parameters

Name	In	Type	Description
pool	body	object	A pool object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
description	body	string	A human-readable description for the resource.
health_monitors	body	array	List of health monitors that are associated with the pool.
health_monitors_status	body	string	The statuses of the health monitors that are associated with the pool.
id	body	string	The ID of the pool.
lb_method	body	string	The load-balancer algorithm, which is round-robin (ROUND_ROBIN), least-connections (LEAST_CONNECTIONS), source IP (SOURCE_IP), and so on, that is used to distribute traffic to the pool members. This value, which must be supported, is dependent on the load-balancer provider. The round-robin algorithm must be supported.
members	body	array	The list of members that belong to the pool.
name	body	string	Human-readable name of the resource.
protocol	body	string	The protocol of the pool, which is TCP, HTTP, or HTTPS.
provider	body	string	The provider name of the load balancer pool.
status	body	string	The status of the pool. Indicates whether the pool is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The subnet on which the members of the pool will be located.
tenant_id	body	string	The ID of the project.
vip_id	body	string	The ID of the virtual IP (VIP)



Name	In	Type	Description
			address.

## Response Example

```
{
  "pool": {
    "admin_state_up": true,
    "description": "",
    "health_monitors": [
      "b7633ade-24dc-4d72-8475-06aa22be5412"
    ],
    "health_monitors_status": [
      {
        "monitor_id": "b7633ade-24dc-4d72-8475-06aa22be5412",
        "status": "ACTIVE",
        "status_description": null
      }
    ],
    "id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "lb_algorithm": "ROUND_ROBIN",
    "members": [
      "cf024846-7516-4e3a-b0fb-6590322c836f"
    ],
    "name": "SuperPool",
    "project_id": "eabfefa3fd1740a88a47ad98e132d238",
    "protocol": "HTTP",
    "provider": "haproxy",
    "status": "PENDING_UPDATE",
    "status_description": null,
    "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "vip_id": "388c739a-6a57-4e74-bc7b-a5cd60248bba"
  }
}
```

## DELETE

/v2.0/lb/pools/{pool\_id}

## Delete pool

Deletes a load balancer pool.

Normal response codes: 204

Error response codes: 401, 404, 409

When a requested load balancer pool has a VIP, Conflict (409) is returned.

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/lb/vips

## List VIPs

Lists VIPs.

The list might be empty.

Normal response codes: 200

Error response codes: 401

Request

Response Parameters

Name	In	Type	Description
vips	body	array	A list of vip objects.
address	body	string	The IP address of the VIP.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the virtual IP (VIP) address.
name	body	string	Human-readable name of the resource.
pool_id	body	string	The ID of the pool which the VIP belongs to.
port_id	body	string	The ID of the VIP port.
protocol	body	string	The protocol of the virtual IP (VIP) address. A valid value is TCP, HTTP, or HTTPS.
protocol_port	body	integer	The TCP or UDP port on which to listen.
session_persistence	body	object	Session persistence parameters for the VIP. Omit the session_persistence parameter to prevent session persistence. When no session persistence is used, the

Name	In	Type	Description
			session_persistence parameter does not appear in the API response.
status	body	string	The status of the VIP. Indicates whether the VIP is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address.
tenant_id	body	string	The ID of the project.

### Response Example

```
{
  "vips": [
    {
      "address": "10.0.0.4",
      "admin_state_up": true,
      "connection_limit": -1,
      "description": "",
      "id": "388c739a-6a57-4e74-bc7b-a5cd60248bba",
      "name": "my-Vip",
      "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
      "port_id": "5328aeea-2988-41c0-b5fe-0fd0660979d3",
      "protocol": "HTTP",
      "protocol_port": 80,
      "session_persistence": null,
      "status": "ACTIVE",
      "status_description": null,
      "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
      "tenant_id": "eabfef3fd1740a88a47ad98e132d238"
    }
  ]
}
```

**POST**

/v2.0/lb/vips

### Create a load balancer VIP

Creates a load balancer VIP.

Normal response codes: 201

Error response codes: 400, 401, 404, 409

When a specified pool already has a VIP, Conflict (409) is returned. When a specified pool or a specified subnet is not found, Not Found (404) is returned.

## Request

Name	In	Type	Description
vip	body	object	A vip object.
address (Optional)	body	string	The IP address of the VIP.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
connection_limit (Optional)	body	integer	The maximum number of connections permitted for this load balancer. Default is infinite.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
pool_id	body	string	The ID of the pool which the VIP belongs to.
protocol	body	string	The protocol of the virtual IP (VIP) address. A valid value is TCP, HTTP, or HTTPS.
protocol_port	body	integer	The TCP or UDP port on which to listen.
session_persistence (Optional)	body	object	Session persistence parameters for the VIP. Omit the session_persistence parameter to prevent session persistence. When no session persistence is used, the session_persistence parameter does not appear in the API response. To clear session persistence for the VIP, set the session_persistence parameter to null in a VIP update request. The default is no session persistence.
subnet_id	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.

## Request Example

```
{
  "vip": {
    "admin_state_up": true,
    "name": "NewVip",
    "pool_id": "105320c3-8416-4997-9c1c-4098b95fdaca",
    "protocol": "HTTP",
    "protocol_port": "80",
    "subnet_id": "0ba2ef27-0054-4b28-a8fa-f215e8079272"
  }
}
```

## Response Parameters

Name	In	Type	Description
vip	body	object	A vip object.
address	body	string	The IP address of the VIP.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the virtual IP (VIP) address.
name	body	string	Human-readable name of the resource.
pool_id	body	string	The ID of the pool which the VIP belongs to.
port_id	body	string	The ID of the VIP port.
protocol	body	string	The protocol of the virtual IP (VIP) address. A valid value is TCP, HTTP, or HTTPS.
protocol_port	body	integer	The TCP or UDP port on which to listen.
session_persistence	body	object	Session persistence parameters for the VIP. Omit the session_persistence parameter to prevent session persistence. When no session persistence is used, the session_persistence parameter does not appear in the API response.
status	body	string	The status of the VIP. Indicates whether the VIP is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "vip": {
    "address": "10.0.0.4",
    "admin_state_up": true,
    "connection_limit": -1,
    "description": "",
    "id": "fa0373e0-9dd4-4ff7-98fc-8cceca9bdb4e",
    "name": "NewVip",
    "pool_id": "105320c3-8416-4997-9c1c-4098b95fdaca",
    "port_id": "0ba4cd9c-edb4-4594-bac4-b68b49d5f04c",
    "protocol": "HTTP",
    "protocol_port": 80,
    "session_persistence": null,
    "status": "PENDING_CREATE",
    "status_description": null,
    "subnet_id": "0ba2ef27-0054-4b28-a8fa-f215e8079272",
    "tenant_id": "e68c3e65e1f34ee9b2357d0fe418a78b"
  }
}
```

**GET**

/v2.0/lb/vips/{vip\_id}

## Show VIP details

Shows details for a VIP.

Normal response codes: 200

Error response codes: 401, 404

## Request

Name	In	Type	Description
vip_id	path	string	The ID for the VIP.

## Response Parameters

Name	In	Type	Description
vip	body	object	A vip object.
address	body	string	The IP address of the VIP.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the virtual IP (VIP) address.

Name	In	Type	Description
name	body	string	Human-readable name of the resource.
pool_id	body	string	The ID of the pool which the VIP belongs to.
port_id	body	string	The ID of the VIP port.
protocol	body	string	The protocol of the virtual IP (VIP) address. A valid value is TCP, HTTP, or HTTPS.
protocol_port	body	integer	The TCP or UDP port on which to listen.
session_persistence	body	object	Session persistence parameters for the VIP. Omit the session_persistence parameter to prevent session persistence. When no session persistence is used, the session_persistence parameter does not appear in the API response.
status	body	string	The status of the VIP. Indicates whether the VIP is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address.
tenant_id	body	string	The ID of the project.

## Response Example

```
{
  "vip": {
    "address": "10.0.0.4",
    "admin_state_up": true,
    "connection_limit": -1,
    "description": "",
    "id": "388c739a-6a57-4e74-bc7b-a5cd60248bba",
    "name": "my-Vip",
    "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "port_id": "5328aeea-2988-41c0-b5fe-0fd0660979d3",
    "protocol": "HTTP",
    "protocol_port": 80,
    "session_persistence": null,
    "status": "ACTIVE",
    "status_description": null,
    "subnet_id": "aa547115-d710-4d6d-bb2c-b038d9c2704b",
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238"
  }
}
```

PUT

/v2.0/lb/vips/{vip\_id}

## Update VIP

Updates a load balancer VIP.

Normal response codes: 200

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
vip_id	path	string	The ID for the VIP.
vip	body	object	A vip object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
connection_limit (Optional)	body	integer	The maximum number of connections permitted for this load balancer. Default is infinite.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
pool_id (Optional)	body	string	The ID of the pool which the VIP belongs to.
session_persistence (Optional)	body	object	Session persistence parameters for the VIP. Omit the session_persistence parameter to prevent session persistence. When no session persistence is used, the session_persistence parameter does not appear in the API response. To clear session persistence for the VIP, set the session_persistence parameter to null in a VIP update request. The default is no session persistence.

### Request Example

```
{
  "vip": {
    "connection_limit": "1000"
  }
}
```



## Response Parameters

Name	In	Type	Description
vip	body	object	A vip object.
address	body	string	The IP address of the VIP.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
connection_limit	body	integer	The maximum number of connections allowed for the VIP. Value is -1 if the limit is not set. Default is infinite.
description	body	string	A human-readable description for the resource.
id	body	string	The ID of the virtual IP (VIP) address.
name	body	string	Human-readable name of the resource.
pool_id	body	string	The ID of the pool which the VIP belongs to.
port_id	body	string	The ID of the VIP port.
protocol	body	string	The protocol of the virtual IP (VIP) address. A valid value is TCP, HTTP, or HTTPS.
protocol_port	body	integer	The TCP or UDP port on which to listen.
session_persistence	body	object	Session persistence parameters for the VIP. Omit the session_persistence parameter to prevent session persistence. When no session persistence is used, the session_persistence parameter does not appear in the API response.
status	body	string	The status of the VIP. Indicates whether the VIP is operational.
status_description	body	string	Human-readable description of the status.
subnet_id	body	string	The ID of the subnet on which to allocate the virtual IP (VIP) address.
tenant_id	body	string	The ID of the project.

Name	In	Type	Description

## Response Example

```
{
  "vip": {
    "address": "10.0.0.4",
    "admin_state_up": true,
    "connection_limit": 1000,
    "description": "",
    "id": "fa0373e0-9dd4-4ff7-98fc-8cceca9bdb4e",
    "name": "NewVip",
    "pool_id": "105320c3-8416-4997-9c1c-4098b95fdaca",
    "port_id": "0ba4cd9c-edb4-4594-bac4-b68b49d5f04c",
    "protocol": "HTTP",
    "protocol_port": 80,
    "session_persistence": null,
    "status": "PENDING_UPDATE",
    "status_description": null,
    "subnet_id": "0ba2ef27-0054-4b28-a8fa-f215e8079272",
    "tenant_id": "e68c3e65e1f34ee9b2357d0fe418a78b"
  }
}
```

## DELETE

/v2.0/lb/vips/{vip\_id}

## Delete VIP

Deletes a load balancer VIP.

Normal response codes: 204

Error response codes: 401, 404

## Request

Name	In	Type	Description
vip_id	path	string	The ID for the VIP.

## Response

There is no body content for the response of a successful DELETE request.

## GET

/v2.0/lb/members

## List members

Lists members.

Normal response codes: 200

Error response codes: 401

## Request

## Response Parameters

Name	In	Type	Description
members	body	array	A list of member objects.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
id	body	string	The ID of the member.
pool_id	body	string	The ID of the pool which the member belongs to.
protocol_port	body	integer	The TCP or UDP port on which to listen.
status	body	string	The status of the member. Indicates whether the member is operational.
status_description	body	string	Human-readable description of the status.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "members": [
    {
      "address": "10.0.1.22",
      "admin_state_up": true,
      "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
      "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
      "protocol_port": 90,
      "status": "ACTIVE",
      "status_description": null,
      "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
      "weight": 1
    }
  ]
}
```

**POST**

/v2.0/lb/members

## Create a load balancer member

Creates a load balancer member.

Normal response codes: 201

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
member	body	object	A member object.
address	body	string	The IP address of the member.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
pool_id	body	string	The ID of the pool which the member belongs to.
protocol_port	body	integer	The TCP or UDP port on which to listen.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
weight (Optional)	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256. The default is 1.

### Request Example

```
{
  "member": {
    "address": "10.0.0.22",
    "admin_state_up": true,
    "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "protocol_port": 90,
    "weight": 1
  }
}
```

### Response Parameters

Name	In	Type	Description
member	body	object	A member object.

Name	In	Type	Description
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
id	body	string	The ID of the member.
pool_id	body	string	The ID of the pool which the member belongs to.
protocol_port	body	integer	The TCP or UDP port on which to listen.
status	body	string	The status of the member. Indicates whether the member is operational.
status_description	body	string	Human-readable description of the status.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "member": {
    "address": "10.0.0.22",
    "admin_state_up": true,
    "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
    "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "protocol_port": 90,
    "status": "ACTIVE",
    "status_description": null,
    "tenant_id": "eabfefaf3fd1740a88a47ad98e132d238",
    "weight": 1
  }
}
```

**GET**

/v2.0/lb/members/{member\_id}

## Show member details

Shows details for a member.

Normal response codes: 200

Error response codes: 401, 404

Request

Name	In	Type	Description
member_id	path	string	The ID for the member.

### Response Parameters

Name	In	Type	Description
member	body	object	A member object.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
id	body	string	The ID of the member.
pool_id	body	string	The ID of the pool which the member belongs to.
protocol_port	body	integer	The TCP or UDP port on which to listen.
status	body	string	The status of the member. Indicates whether the member is operational.
status_description	body	string	Human-readable description of the status.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

### Response Example

```
{
  "member": {
    "address": "10.0.1.22",
    "admin_state_up": true,
    "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
    "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "protocol_port": 90,
```

```

    "status": "ACTIVE",
    "status_description": null,
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "weight": 1
  }
}

```

**PUT**

/v2.0/lb/members/{member\_id}

## Update member

Updates a load balancer member.

Normal response codes: 200

Error response codes: 400, 401, 404

### Request

Name	In	Type	Description
member_id	path	string	The ID for the member.
member	body	object	A member object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
pool_id (Optional)	body	string	The ID of the pool which the member belongs to.
weight (Optional)	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256. The default is 1.

### Request Example

```

{
  "member": {
    "weight": 5
  }
}

```

### Response Parameters

Name	In	Type	Description
member	body	object	A member object.
address	body	string	The IP address of the member.
admin_state_up	body	boolean	The administrative state of the resource,

Name	In	Type	Description
			which is up (true) or down (false).
id	body	string	The ID of the member.
pool_id	body	string	The ID of the pool which the member belongs to.
protocol_port	body	integer	The TCP or UDP port on which to listen.
status	body	string	The status of the member. Indicates whether the member is operational.
status_description	body	string	Human-readable description of the status.
tenant_id	body	string	The ID of the project.
weight	body	integer	The weight of a member determines the portion of requests or connections it services compared to the other members of the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2. A value of 0 means the member does not participate in load- balancing but still accepts persistent connections. A valid value is from 0 to 256.

## Response Example

```
{
  "member": {
    "address": "10.0.0.22",
    "admin_state_up": true,
    "id": "cf024846-7516-4e3a-b0fb-6590322c836f",
    "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
    "protocol_port": 90,
    "status": "ACTIVE",
    "status_description": null,
    "tenant_id": "eabfefaf3fd1740a88a47ad98e132d238",
    "weight": 5
  }
}
```

## DELETE

/v2.0/lb/members/{member\_id}

## Delete member

Deletes a load balancer member.

Normal response codes: 204

Error response codes: 401, 404



## Request

Name	In	Type	Description
member_id	path	string	The ID for the member.

## Response

There is no body content for the response of a successful DELETE request.

### GET

/v2.0/lb/health\_monitors

## List health monitors

Lists health monitors.

Normal response codes: 200

Error response codes: 401

## Request

### Response Parameters

Name	In	Type	Description
health_monitors	body	array	A list of health_monitor objects.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
id	body	string	The ID of the associated

Name	In	Type	Description
			health monitor.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pools	body	array	List of pools that are associated with the health monitor.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitors": [
    {
      "admin_state_up": true,
      "delay": 1,
      "expected_codes": "200,201,202",
      "http_method": "GET",
      "id": "b7633ade-24dc-4d72-8475-06aa22be5412",
      "max_retries": 5,
      "pools": [
        {
          "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
          "status": "ACTIVE",
          "status_description": null
        }
      ],
      "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
      "timeout": 1,
      "type": "HTTP",
      "url_path": "/index.html"
    }
  ]
}
```

**POST**

/v2.0/lb/health\_monitors

## Create a load balancer health monitor

Creates a load balancer health monitor.

Normal response codes: 201

Error response codes: 400, 401

### Request

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes (Optional)	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method (Optional)	body	string	The HTTP method that the monitor uses for requests. The default is GET.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times

Name	In	Type	Description
			out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path (Optional)	body	string	The HTTP path of the request sent by the monitor to test the health of a member. A valid value is a string that begins with a forward slash (/).

### Request Example

```
{
  "health_monitor": {
    "admin_state_up": true,
    "delay": 1,
    "expected_codes": "200,201,202",
    "http_method": "GET",
    "max_retries": 5,
    "pool_id": "74aa2010-a59f-4d35-a436-60a6da882819",
    "timeout": 1,
    "type": "HTTP",
    "url_path": "/index.html"
  }
}
```

### Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"> <li>A single value, such as 200</li> <li>A list, such as 200, 202</li> <li>A range, such as 200-204</li> </ul>

Name	In	Type	Description
			The default is 200.
http_method	body	string	The HTTP method that the monitor uses for requests.
id	body	string	The ID of the associated health monitor.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pools	body	array	List of pools that are associated with the health monitor.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitor": {
    "admin_state_up": true,
    "delay": 1,
    "expected_codes": "200,201,202",
    "http_method": "GET",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412",
    "max_retries": 5,
    "pools": [],
    "tenant_id": "eabfef3fd1740a88a47ad98e132d238",
    "timeout": 1,
    "type": "HTTP",
    "url_path": "/index.html"
  }
}
```

**GET**  
/v2.0/lb/health\_monitors/{health\_monitor\_id}

## Show health monitor details

Shows details for a health monitor.

Normal response codes: 200

Error response codes: 401, 404

Request

Name	In	Type	Description
health_monitor_id	path	string	The ID for the health monitor.

### Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
id	body	string	The ID of the associated health monitor.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pools	body	array	List of pools that are associated with the health

Name	In	Type	Description
			monitor.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default is /.

## Response Example

```
{
  "health_monitor": {
    "admin_state_up": true,
    "delay": 1,
    "expected_codes": "200,201,202",
    "http_method": "GET",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412",
    "max_retries": 5,
    "pools": [
      {
        "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
        "status": "ACTIVE",
        "status_description": null
      }
    ],
    "tenant_id": "eabfefaf3fd1740a88a47ad98e132d238",
    "timeout": 1,
    "type": "HTTP",
    "url_path": "/index.html"
  }
}
```

**PUT**

/v2.0/lb/health\_monitors/{health\_monitor\_id}

## Update health monitor

Updates a load balancer health monitor.

Normal response codes: 200

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
health_monitor_id	path	string	The ID for the health monitor.
health_monitor	body	object	A health_monitor object.
admin_state_up (Optional)	body	boolean	The administrative state of the resource, which is up (true) or down (false). Default is true.
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes (Optional)	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"><li>• A single value, such as 200</li><li>• A list, such as 200, 202</li><li>• A range, such as 200-204</li></ul> <p>The default is 200.</p>
http_method (Optional)	body	string	The HTTP method that the monitor uses for requests. The default is GET.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
url_path (Optional)	body	string	The HTTP path of the request sent by the monitor to test the health of a member. A valid value is a string that begins with a forward slash (/).

## Request Example

```
{
  "health_monitor": {
    "admin_state_up": false,
    "delay": 2,
    "expected_codes": "200",
    "http_method": "POST",
    "max_retries": 2,
    "timeout": 2,
    "url_path": "/page.html"
  }
}
```

## Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.



Name	In	Type	Description
admin_state_up	body	boolean	The administrative state of the resource, which is up (true) or down (false).
delay	body	integer	The time, in seconds, between sending probes to members.
expected_codes	body	string	<p>The list of HTTP status codes expected in response from the member to declare it healthy. Specify one of the following values:</p> <ul style="list-style-type: none"> <li>• A single value, such as 200</li> <li>• A list, such as 200, 202</li> <li>• A range, such as 200-204</li> </ul> <p>The default is 200.</p>
http_method	body	string	The HTTP method that the monitor uses for requests.
id	body	string	The ID of the associated health monitor.
max_retries	body	integer	The number of allowed connection failures before changing the status of the member to INACTIVE. A valid value is from 1 to 10.
pools	body	array	List of pools that are associated with the health monitor.
tenant_id	body	string	The ID of the project.
timeout	body	integer	The maximum time, in seconds, that a monitor waits to connect before it times out. This value must be less than the delay value.
type	body	string	The type of probe sent by the load balancer to verify the member state. A valid value is PING, TCP, HTTP, or HTTPS.
url_path	body	string	The HTTP path of the request sent by the monitor to test the health of a member. Must be a string that begins with a forward slash (/). The default

Name	In	Type	Description
			is /.

## Response Example

```
{
  "health_monitor": {
    "admin_state_up": false,
    "delay": 2,
    "expected_codes": "200",
    "http_method": "POST",
    "id": "b7633ade-24dc-4d72-8475-06aa22be5412",
    "max_retries": 2,
    "pools": [
      {
        "pool_id": "5a9a3e9e-d1aa-448e-af37-a70171f2a332",
        "status": "ACTIVE",
        "status_description": null
      }
    ],
    "tenant_id": "eabfefa3fd1740a88a47ad98e132d238",
    "timeout": 2,
    "type": "HTTP",
    "url_path": "/page.html"
  }
}
```

## DELETE

/v2.0/lb/health\_monitors/{health\_monitor\_id}

## Delete health monitor

Deletes a load balancer health monitor.

Normal response codes: 204

Error response codes: 401, 404, 409

## Request

Name	In	Type	Description
health_monitor_id	path	string	The ID for the health monitor.

## Response

There is no body content for the response of a successful DELETE request.

## POST

/v2.0/lb/pools/{pool\_id}/health\_monitors

## Associate health monitor with pool

Associates a health monitor with a pool.

Normal response codes: 201

Error response codes: 400, 401, 404

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
health_monitor	body	object	A health_monitor object.
id	body	string	The ID of the associated health monitor.

## Request Example

```
{
  "health_monitor": {
    "id": "b624decf-d5d3-4c66-9a3d-f047e7786181"
  }
}
```

## Response Parameters

Name	In	Type	Description
health_monitor	body	object	A health_monitor object.

## Response Example

```
{
  "health_monitor": {}
}
```

## DELETE

/v2.0/lb/pools/{pool\_id}/health\_monitors/{health\_monitor\_id}

## Disassociate health monitor from pool

Disassociates a health monitor from a pool.

Normal response codes: 201

Error response codes: 401, 404

## Request

Name	In	Type	Description
pool_id	path	string	The ID for the pool.
health_monitor_id	path	string	The ID for the health monitor.

## Response

There is no body content for the response of a successful DELETE request.

# Logging Resource (networking-midonet)

## Logging Resources

### Note

Currently this extension `logging-resource` is only available for `networking-midonet`.

Lists, shows information for, creates, updates and deletes logging resources.

### GET

`/v2.0/logging/logging_resources`

### List Logging Resources

Lists logging resources.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

### Request

Name	In	Type	Description
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

### Response

Name	In	Type	Description
logging_resources	body	array	A list of <code>logging_resource</code> objects.
id	body	string	The ID of the logging resource.
tenant_id	body	string	The ID of the project.

Name	In	Type	Description
project_id	body	string	The ID of the project.
firewall_logs	body	array	A list of firewall_log objects.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
enabled	body	boolean	Indicates whether this logging resource is enabled or disabled.

## Response Example

```
{
  "logging_resources": [
    {
      "description": "my log",
      "enabled": true,
      "firewall_logs": [],
      "id": "13b64f3c-20af-4741-b230-658ab7d5b257",
      "name": "log",
      "project_id": "8d018258316e4f22890561e8780c85bb",
      "tenant_id": "8d018258316e4f22890561e8780c85bb"
    },
    {
      "description": "my log2",
      "enabled": true,
      "firewall_logs": [],
      "id": "335c7b7d-c4a9-423a-9c24-9f4982f31e24",
      "name": "log2",
      "project_id": "8d018258316e4f22890561e8780c85bb",
      "tenant_id": "8d018258316e4f22890561e8780c85bb"
    }
  ]
}
```

## POST

/v2.0/logging/logging\_resources

## Create Logging Resource

Creates a logging resource.

Normal response codes: 200

Error response codes: 400, 401, 403

## Request

Name	In	Type	Description
logging_resource	body	object	A logging_resource object.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only

Name	In	Type	Description
			administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
name (Optional)	body	string	Human-readable name of the resource. Default is an empty string.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
enabled (Optional)	body	boolean	Indicates whether this logging resource is enabled or disabled. Default is false.

### Request Example

```
{
  "logging_resource": {
    "description": "my log",
    "enabled": true,
    "name": "log"
  }
}
```

### Response

Name	In	Type	Description
logging_resource	body	object	A logging_resource object.
id	body	string	The ID of the logging resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
firewall_logs	body	array	A list of firewall_log objects.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
enabled	body	boolean	Indicates whether this logging

Name	In	Type	Description
			resource is enabled or disabled.

## Response Example

```
{
  "logging_resource": {
    "description": "my log",
    "enabled": true,
    "firewall_logs": [],
    "id": "13b64f3c-20af-4741-b230-658ab7d5b257",
    "name": "log",
    "project_id": "8d018258316e4f22890561e8780c85bb",
    "tenant_id": "8d018258316e4f22890561e8780c85bb"
  }
}
```

**GET**

/v2.0/logging/logging\_resources/{logging\_resource\_id}

## Show Logging Resource Details

Shows details for a logging resource.

Use the `fields` query parameter to control which fields are returned in the response body. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response

Name	In	Type	Description
logging_resource	body	object	A <code>logging_resource</code> object.
id	body	string	The ID of the logging resource.
tenant_id	body	string	The ID of the project.

Name	In	Type	Description
project_id	body	string	The ID of the project.
firewall_logs	body	array	A list of firewall_log objects.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
enabled	body	boolean	Indicates whether this logging resource is enabled or disabled.

## Response Example

```
{
  "logging_resource": {
    "description": "my log",
    "enabled": true,
    "firewall_logs": [
      {
        "description": "",
        "firewall_id": "682cfe44-5fcf-4c16-982e-1176493f6825",
        "fw_event": "ALL",
        "id": "1ee6fea7-c294-418e-9b97-06db48e3f3d5",
        "logging_resource_id": "13b64f3c-20af-4741-b230-658ab7d5b257",
        "project_id": "8d018258316e4f22890561e8780c85bb",
        "tenant_id": "8d018258316e4f22890561e8780c85bb"
      }
    ],
    "id": "13b64f3c-20af-4741-b230-658ab7d5b257",
    "name": "log",
    "project_id": "8d018258316e4f22890561e8780c85bb",
    "tenant_id": "8d018258316e4f22890561e8780c85bb"
  }
}
```

**PUT**

/v2.0/logging/logging\_resources/{logging\_resource\_id}

## Update Logging Resource

Updates a logging resource.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.
logging_resource	body	object	A logging_resource object.



Name	In	Type	Description
name (Optional)	body	string	Human-readable name of the resource.
description (Optional)	body	string	A human-readable description for the resource.
enabled (Optional)	body	boolean	Indicates whether this logging resource is enabled or disabled.

### Request Example

```
{
  "logging_resource": {
    "description": "my log2",
    "enabled": false
  }
}
```

### Response

Name	In	Type	Description
logging_resource	body	object	A logging_resource object.
id	body	string	The ID of the logging resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
firewall_logs	body	array	A list of firewall_log objects.
name	body	string	Human-readable name of the resource.
description	body	string	A human-readable description for the resource.
enabled	body	boolean	Indicates whether this logging resource is enabled or disabled.

### Response Example

```
{
  "logging_resource": {
    "description": "my log2",
    "enabled": false,
    "firewall_logs": [
      {
        "description": "",
        "firewall_id": "682cfe44-5fcf-4c16-982e-1176493f6825",
        "fw_event": "ALL",
        "id": "1ee6fea7-c294-418e-9b97-06db48e3f3d5",
        "logging_resource_id": "335c7b7d-c4a9-423a-9c24-9f4982f31e24",
        "project_id": "8d018258316e4f22890561e8780c85bb",
        "tenant_id": "8d018258316e4f22890561e8780c85bb"
      }
    ]
  }
}
```

```
    "id": "335c7b7d-c4a9-423a-9c24-9f4982f31e24",  
    "name": "log2",  
    "project_id": "8d018258316e4f22890561e8780c85bb",  
    "tenant_id": "8d018258316e4f22890561e8780c85bb"  
  }  
}
```

## DELETE

/v2.0/logging/logging\_resources/{logging\_resource\_id}

### Delete Logging Resource

Deletes a logging resource.

Normal response codes: 202

Error response codes: 400, 401, 403, 404

#### Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.

#### Response

There is no body content for the response of a successful DELETE request.

## Firewall Logs

### Note

Currently this extension `logging-resource` is only available for `networking-midonet`.

Lists, shows information for, creates, updates and deletes firewall logs.

## GET

/v2.0/logging/logging\_resources/{logging\_resource\_id}/firewall\_logs

### List Firewall Logs

Lists firewall logs.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 401, 403

## Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.
fields (Optional)	query	string	The fields that you want the server to return. If no fields query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using fields parameter, the API returns only the requested set of attributes. fields parameter can be specified multiple times. For example, if you specify fields=id&fields=name in the request URL, only id and name attributes will be returned.

## Response

Name	In	Type	Description
firewall_logs	body	array	A list of firewall_log objects.
logging_resource_id	body	string	The ID of the logging resource.
id	body	string	The ID of the firewall log resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
description	body	string	A human-readable description for the resource.
fw_event	body	string	Type of firewall events to log. ACCEPT, DROP, or ALL.
firewall_id	body	string	The ID of the FWaaS v1 firewall.

## Response Example

```
{
  "firewall_logs": [
    {
      "description": "my firewall log 2",
      "firewall_id": "a6564146-f8b3-49c3-add1-fb213455d5a8",
      "fw_event": "ACCEPT",
      "id": "3969b708-d600-4343-93b9-01645f8e9a8a",
      "logging_resource_id": "13b64f3c-20af-4741-b230-658ab7d5b257",
      "project_id": "8d018258316e4f22890561e8780c85bb",
      "tenant_id": "8d018258316e4f22890561e8780c85bb"
    },
    {
      "description": "my firewall log",
      "firewall_id": "a6564146-f8b3-49c3-add1-fb213455d5a8",
      "fw_event": "DROP",
      "id": "deb19331-e5d5-4a80-a37f-5e5ad407b353",
      "logging_resource_id": "13b64f3c-20af-4741-b230-658ab7d5b257",
      "project_id": "8d018258316e4f22890561e8780c85bb",
      "tenant_id": "8d018258316e4f22890561e8780c85bb"
    }
  ]
}
```

```
}
```

**POST**

/v2.0/logging/logging\_resources/{logging\_resource\_id}/firewall\_logs

## Create Firewall Log

Creates a firewall log.

Normal response codes: 200

Error response codes: 400, 401, 403

### Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.
firewall_log	body	object	A firewall_log object.
tenant_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
project_id (Optional)	body	string	The ID of the project that owns the resource. Only administrative users can specify a project ID other than their own. You cannot change this value through authorization policies.
description (Optional)	body	string	A human-readable description for the resource. Default is an empty string.
fw_event (Optional)	body	string	Type of firewall events to log. ACCEPT, DROP, or ALL. Default is ALL.
firewall_id	body	string	The ID of the FWaaS v1 firewall.

### Request Example

```
{
  "firewall_log": {
    "description": "my firewall log",
    "firewall_id": "a6564146-f8b3-49c3-add1-fb213455d5a8",
    "fw_event": "DROP"
  }
}
```

## Response

Name	In	Type	Description
firewall_log	body	object	A firewall_log object.
logging_resource_id	body	string	The ID of the logging resource.
id	body	string	The ID of the firewall log resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
description	body	string	A human-readable description for the resource.
fw_event	body	string	Type of firewall events to log. ACCEPT, DROP, or ALL.
firewall_id	body	string	The ID of the FWaaS v1 firewall.

## Response Example

```
{
  "firewall_log": {
    "description": "my firewall log",
    "firewall_id": "a6564146-f8b3-49c3-add1-fb213455d5a8",
    "fw_event": "DROP",
    "id": "deb19331-e5d5-4a80-a37f-5e5ad407b353",
    "logging_resource_id": "13b64f3c-20af-4741-b230-658ab7d5b257",
    "project_id": "8d018258316e4f22890561e8780c85bb",
    "tenant_id": "8d018258316e4f22890561e8780c85bb"
  }
}
```

### GET

/v2.0/logging/logging\_resources/{logging\_resource\_id}/firewall\_logs/{firewall\_log\_id}

## Show Firewall Log Details

Shows details for a firewall log.

Use the `fields` query parameter to control which fields are returned in the response body. Additionally, you can filter results by using query string parameters. For information, see [Filtering and Column Selection](#).

Normal response codes: 200

Error response codes: 400, 401, 403, 404

## Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging

Name	In	Type	Description
			resource.
firewall_log_id	path	string	The ID of the firewall log resource.
fields (Optional)	query	string	The fields that you want the server to return. If no <code>fields</code> query parameter is specified, the networking API returns all attributes allowed by the policy settings. By using <code>fields</code> parameter, the API returns only the requested set of attributes. <code>fields</code> parameter can be specified multiple times. For example, if you specify <code>fields=id&amp;fields=name</code> in the request URL, only <code>id</code> and <code>name</code> attributes will be returned.

## Response

Name	In	Type	Description
firewall_log	body	object	A <code>firewall_log</code> object.
logging_resource_id	body	string	The ID of the logging resource.
id	body	string	The ID of the firewall log resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
description	body	string	A human-readable description for the resource.
fw_event	body	string	Type of firewall events to log. ACCEPT, DROP, or ALL.
firewall_id	body	string	The ID of the FWaaS v1 firewall.

## Response Example

```
{
  "firewall_log": {
    "description": "my firewall log 3",
    "firewall_id": "a6564146-f8b3-49c3-add1-fb213455d5a8",
    "fw_event": "ALL",
    "id": "3969b708-d600-4343-93b9-01645f8e9a8a",
    "logging_resource_id": "13b64f3c-20af-4741-b230-658ab7d5b257",
    "project_id": "8d018258316e4f22890561e8780c85bb",
    "tenant_id": "8d018258316e4f22890561e8780c85bb"
  }
}
```

PUT

/v2.0/logging/logging\_resources/{logging\_resource\_id}/firewall\_logs/{firewall\_log\_id}

## Update Firewall Log

Updates a firewall log.

Normal response codes: 200

Error response codes: 400, 401, 403, 404

### Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.
firewall_log_id	path	string	The ID of the firewall log resource.
firewall_log	body	object	A firewall_log object.
description (Optional)	body	string	A human-readable description for the resource.
fw_event (Optional)	body	string	Type of firewall events to log. ACCEPT, DROP, or ALL.

### Request Example

```
{
  "firewall_log": {
    "description": "my firewall log 3",
    "fw_event": "ALL"
  }
}
```

### Response

Name	In	Type	Description
firewall_log	body	object	A firewall_log object.
logging_resource_id	body	string	The ID of the logging resource.
id	body	string	The ID of the firewall log resource.
tenant_id	body	string	The ID of the project.
project_id	body	string	The ID of the project.
description	body	string	A human-readable description for the resource.
fw_event	body	string	Type of firewall events to log.

Name	In	Type	Description
			ACCEPT, DROP, or ALL.
firewall_id	body	string	The ID of the FWaaS v1 firewall.

### Response Example

```
{
  "firewall_log": {
    "description": "my firewall log 3",
    "firewall_id": "a6564146-f8b3-49c3-add1-fb213455d5a8",
    "fw_event": "ALL",
    "id": "3969b708-d600-4343-93b9-01645f8e9a8a",
    "logging_resource_id": "13b64f3c-20af-4741-b230-658ab7d5b257",
    "project_id": "8d018258316e4f22890561e8780c85bb",
    "tenant_id": "8d018258316e4f22890561e8780c85bb"
  }
}
```

### DELETE

/v2.0/logging/logging\_resources/{logging\_resource\_id}/firewall\_logs/{firewall\_log\_id}

### Delete Firewall Log

Deletes a firewall log.

Normal response codes: 202

Error response codes: 400, 401, 403, 404

### Request

Name	In	Type	Description
logging_resource_id	path	string	The ID of the logging resource.
firewall_log_id	path	string	The ID of the firewall log resource.

### Response

There is no body content for the response of a successful DELETE request.

# Router interface floating IP (networking-midonet)

## Router Interface floating IP

### Note



Currently this extension `router-interface-fip` is only available for `networking-midonet`.

This extension `router-interface-fip` indicates the ability to associate floating IPs to internal interfaces of a router. (Without this extension, floating IPs can be associated only to the gateway interface of a router.)

This extension does not introduce any resources or attributes.

## FIP64 (networking-midonet)

### FIP64

#### Note

Currently this extension `fip64` is only available for `networking-midonet`.

This extension `fip64` provides [NAT64](#) functionality by allowing to associate IPv6 floating IPs to IPv4 fixed IPs. (Without this extension, floating IPs are limited to IPv4.)

This extension does not introduce any resources or attributes.





