

# SDN网络服务

## API参考

产品版本: v7.0.1

发布日期: 2026-08-18

# 目录

|                        |     |
|------------------------|-----|
| 1 API参考 .....          | 1   |
| 1.1 API简介 .....        | 1   |
| 1.2 调用方式 .....         | 3   |
| 1.3 网络 .....           | 9   |
| 1.4 子网 .....           | 26  |
| 1.5 虚拟网卡 .....         | 42  |
| 1.6 QoS策略 .....        | 81  |
| 1.7 路由器 .....          | 93  |
| 1.8 策略路由 .....         | 122 |
| 1.9 安全组 .....          | 137 |
| 1.10 网络ACL/网关防火墙 ..... | 163 |
| 1.11 NAT网关 .....       | 207 |
| 1.12 公网IP .....        | 243 |
| 1.13 发布记录 .....        | 256 |



# 1 API参考

## 1.1 API简介

欢迎使用API文档，如果您熟悉网络服务协议和一种以上编程语言，推荐您调用API管理您的资源和开发自己的应用程序。本文档提供了API的描述、语法、参数说明及示例等内容。在调用API之前，请确保已经充分了解相关术语，详细信息请参见下表。

| 术语          | 说明                                                                                                                                                                                                                                                                                               |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 网络          | 网络与现实世界的交换机 / 路由器 / 服务器 / 连线组成的基础设施网络类似，创建网络后，用户可以在网络内创建子网，创建云主机时选择网络，组建服务器集群。我们提供的基础网络包含共享网络和外部网络，创建在共享网络上的云主机处于同一个网络内，通过安全组保障云主机访问安全。外部网络主要用于公网IP地址的分配。用户可以为项目创建内部网络，并在内部网络中创建子网。如同在物理网络上通过交换机将服务器连接到一起的局域网，服务器通过交换机连接到子网中。不同的内部网络之间是完全隔离的，因此不同的网络中可以配置相同的IP地址而不会产生冲突。同一个网络内可以创建多个子网，以适应业务的需求。 |
| 子网          | 子网是一个IPv4或IPv6的地址段。每个子网都需要定义IP地址的范围和掩码。子网和网络是多对1的关系，一个子网只能属于某个网络；一个网络可以有多个子网，这些子网可以是不同的IP段，但不能重叠，但如果子网在不同的网络中，网段是可以重叠的。                                                                                                                                                                          |
| 虚拟网卡        | 虚拟网卡可以看做是一个虚拟网口。虚拟网卡上定义了MAC地址和IP地址。虚拟网卡和子网是多对1的关系。一个虚拟网卡必须属于某个子网，一个子网可以有多个虚拟网卡。                                                                                                                                                                                                                  |
| QoS策略       | QoS（Quality of Service，服务质量），可以用于设置虚拟网卡的带宽限制。                                                                                                                                                                                                                                                    |
| 安全组         | 安全组提供网络间的访问控制功能，通过安全组中的安全组规则对当前项目中的网络流量进行过滤。安全组规则支持多种网络协议。                                                                                                                                                                                                                                       |
| 网络ACL/网关防火墙 | 网络ACL提供子网级别的访问控制策略，网关防火墙提供路由器北向网络的访问控制策略。                                                                                                                                                                                                                                                        |

| 术语   | 说明                                                                                                                          |
|------|-----------------------------------------------------------------------------------------------------------------------------|
| 路由器  | 用户创建路由器，为不同的子网提供三层路由，从而让子网内的云主机与其他子网的云主机互联互通。也可以将用户创建的内部网络连接到外部网络，让内部网络的云主机访问Internet。路由器配置网关后，还可以为内网的云主机做端口转发，以节约公网IP地址资源。 |
| 公网IP | 独立的IP地址资源，用户可以将申请的公网IP绑定到自己的云主机上，之后便可从外部网络通过公网IP来访问云主机提供的服务。                                                                |
| 网络拓扑 | 展示用户当前所在项目的网络结构图。点击各个设备可以展示详细配置。                                                                                            |
| 云主机  | 运行在云环境上的虚拟机，相当于数据中心的一台物理服务器。用户可以通过选择合适的CPU / 内存 / 操作系统磁盘空间，网络，安全组等配置创建云主机。                                                  |
| 部门   | 部门是云平台中用户权限的一个划分层级，用户不能横跨多个部门。                                                                                              |
| 项目   | 项目是定义资源所有权的基本单元，可理解为租户。所有资源（如云主机等）都要隶属于某个项目中。项目必须隶属于一个部门。项目名称在单个部门中的管理范围内是唯一的，但在整个云平台中可以不唯一。                                |
| 用户   | 用户可以被云管理员、部门管理员创建。用户通过登录后，可以操作云平台提供的各项资源，如云主机 / 云硬盘等。                                                                       |

## 1.2 调用方式

### 请求结构

API支持基于URI发起HTTP/HTTPS GET请求。请求参数需要包含在URI中。本文列举了GET请求中的结构解释，并以云主机的服务接入地址为例进行了说明。

#### 结构示例

以下为一条未编码的URI请求示例：`http://cloud.com/v1/{project_id}/servers` 在本示例中：

- `http` 指定了请求通信协议
- `cloud.com` 指定了服务接入地址
- `/v1/{project_id}/servers` 为资源路径，也即API访问路径

#### 通信协议

支持HTTP或HTTPS协议请求通信。为了获得更高的安全性，推荐您使用HTTPS协议发送请求。涉及敏感数据时，如用户密码和SSH密钥对，推荐使用HTTPS协议。

#### 服务网址

调用本文档所列举的API时均需使用OpenStack身份服务进行身份验证。他们还需要一个从“compute”类型的标识符提取出来的“service URI”。这将是根URI，将添加下面的每个调用来构建一个完整的路径。例如，如果“service URI”是 `http://mycompute.pvt/compute/v2.1`，那么“/servers”的完整API调用是

`http://mycompute.pvt/compute/v2.1/servers`。根据部署计算服务网址可能是http或https，自定义端口，自定义路径，并包含您的租户ID。要知道您的部署网址的唯一方法是通过使用服务目录。计算URI不应该被硬编码在应用程序中，即使他们只希望在单一地点工作。应始终从身份令牌中发现。因此，对于本文件的其余部分，我们将使用短针，其中“GET /servers”的真正含义“GET your\_compute\_service\_URI/servers”。

#### 请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

| 方法 | 说明 |
|----|----|
|----|----|

| 方法     | 说明                                                                                             |
|--------|------------------------------------------------------------------------------------------------|
| GET    | 从服务端读取指定资源的所有信息，包括数据内容和元数据（Metadata）信息，其中元数据在响应头（Response Header）中返回，数据内容在响应体（Response Body）中。 |
| PUT    | 向指定的资源上传数据内容和元数据信息。如果资源已经存在，那么新上传的数据将覆盖之前的内容。                                                  |
| POST   | 向指定的资源上传数据内容。与PUT操作相比，POST的主要区别在于POST一般用来向原有的资源添加信息，而不是替换原有的内容：POST所指的资源一般是处理请求的服务，或是能够处理多块数据。 |
| DELETE | 请求服务器删除指定资源，如删除对象等。                                                                            |
| HEAD   | 仅从服务端读取指定资源的元数据信息。                                                                             |

## 字符编码

请求及返回结果都使用UTF-8字符集编码。

## 公共参数

公共参数是用于标识用户和接口签名的参数，如非必要，在每个接口单独的接口文档中不再对这些参数进行说明，但每次请求均需要携带这些参数，才能正常发起请求。

### 公共请求参数

| 名称   | 类型     | 是否必选               | 描述                                                                    |
|------|--------|--------------------|-----------------------------------------------------------------------|
| Host | String | 否（使用AK/SK认证时该字段必选） | 请求的服务器信息，从服务API的URI中获取。值为hostname[:port]。端口缺省时使用默认的端口，https的默认端口为443。 |

| 名称             | 类型     | 是否必选               | 描述                                                                                                        |
|----------------|--------|--------------------|-----------------------------------------------------------------------------------------------------------|
| Content-Type   | String | 是                  | 消息体的类型（格式）。推荐用户使用默认值application/json，有其他取值时会在具体接口中专门说明。                                                   |
| Content-Length | String | 否                  | 请求body长度，单位为Byte。                                                                                         |
| X-Project-Id   | String | 否                  | project id，项目编号。                                                                                          |
| X-Auth-Token   | String | 否（使用Token认证时该字段必选） | 用户Token。用户Token也就是调用获取用户Token接口的响应值，该接口是唯一不需要认证的接口。请求响应成功后在响应消息头（Headers）中包含的“X-Subject-Token”的值即为Token值。 |

## 公共返回参数

| 参数名称      | 参数类型   | 描述                       |
|-----------|--------|--------------------------|
| RequestId | String | 请求ID。无论调用接口成功与否，都会返回该参数。 |

## 签名机制

调用接口的认证方式为Token认证，通过Token认证通用请求。Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限。Token可通过调用获取用户Token接口获取，调用本服务API需要project级别的Token，即调用获取用户Token接口时，请求body中 `auth.scope` 的取值需要选择 `project`，如下所示：

```
{
  "auth": {
    "scope": {
      "project": {
        "domain": {
          "name": "Default"
        }
      }
    }
  }
}
```

```
    },
    "name": "admin"
  }
},
"identity": {
  "password": {
    "user": {
      "password": "devstacker",
      "id": "858634b407e845f14b02bcf369225dcd0"
    }
  },
  "methods": ["password"]
}
}
```

获取Token后，再调用其他接口时，您需要在请求消息头中添加 `X-Auth-Token`，其值即为 `Token`。例如Token值为“ABCDEFJ...”，则调用接口时将 `X-Auth-Token: ABCDEFJ....` 加到请求消息头即可，如下所示：

```
POST https://iam.cn-north-1.mycloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

## 返回结果

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态。为了便于查看和美观，API 文档返回示例均有换行和缩进等处理，实际返回结果无换行和缩进处理。

### 正确返回结果

接口调用成功后会返回接口返回参数和请求 ID，我们称这样的返回为正常返回。HTTP 状态码为 2xx。以云主机的接口创建云主机（POST /v1/{project\_id}/servers）为例，若调用成功，其可能的返回如下：

```
{
  "error": {
    "OS-DCF:diskConfig": "AUTO",
    "adminPass": "6NpUwoz2QDRN",
```

```
    "id": "f5dc173b-6804-445a-a6d8-c705dad5b5eb",
    "links": [
      {
        "href":
"http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/f5
dc173b-6804-445a-a6d8-c705dad5b5eb",
        "rel": "self"
      },
      {
        "href":
"http://openstack.example.com/6f70656e737461636b20342065766572/servers/f5dc1
73b-6804-445a-a6d8-c705dad5b5eb",
        "rel": "bookmark"
      }
    ],
    "security_groups": [
      {
        "name": "default"
      }
    ]
  }
}
```

## 错误返回结果

接口调用出错后，会返回错误码、错误信息和请求 ID，我们称这样的返回为异常返回。HTTP 状态码为 4xx 或者 5xx。

```
{
  "error": {
    "message": "The request you have made requires authentication.",
    "code": 401,
    "title": "Unauthorized"
  }
}
```

## 公共错误码



| http状态码 | Error Message                 | 说明                           |
|---------|-------------------------------|------------------------------|
| 300     | multiple choices              | 被请求的资源存在多个可供选择的响应。           |
| 400     | Bad Request                   | 服务器未能处理请求。                   |
| 401     | Unauthorized                  | 被请求的页面需要用户名和密码。              |
| 403     | Forbidden                     | 对被请求页面的访问被禁止。                |
| 404     | Not Found                     | 服务器无法找到被请求的页面。               |
| 405     | Method Not Allowed            | 请求中指定的方法不被允许。                |
| 406     | Not Acceptable                | 服务器生成的响应无法被客户端所接受。           |
| 407     | Proxy Authentication Required | 用户必须首先使用代理服务器进行验证，这样请求才会被处理。 |
| 408     | Request Timeout               | 请求超出了服务器的等待时间。               |
| 409     | Conflict                      | 由于冲突，请求无法被完成。                |
| 500     | Internal Server Error         | 请求未完成。服务异常。                  |
| 501     | Not Implemented               | 请求未完成。服务器不支持所请求的功能。          |
| 502     | Bad Gateway                   | 请求未完成。服务器从上游服务器收到一个无效的响应。    |
| 503     | Service Unavailable           | 请求未完成。系统暂时异常。                |
| 504     | Gateway Timeout               | 网关超时。                        |

## 1.3 网络

### 网络

#### 列举网络

##### 功能介绍

列出租户有权访问的网络。除具有管理权限的用户提交列举子网的请求外，默认策略是只返回提交请求的用户所拥有的网络。

##### URI

```
GET /v2.0/networks
```

##### 请求消息

无

##### 响应消息

| 参数                    | 参数类型    | 描述                                   |
|-----------------------|---------|--------------------------------------|
| id                    | string  | 网络ID。                                |
| tenant_id             | string  | 项目ID。                                |
| project_id            | string  | 项目ID。                                |
| name                  | string  | 资源的人类可读名称。                           |
| admin_state_up        | boolean | 该网络是否启动。                             |
| port_security_enabled | boolean | 该网络是否开启网络安全防护功能。如果使用安全组功能，此配置项一定为打开。 |
| mtu                   | int     | 该网络数据包最大传输单元大小。                      |
| shared                | boolean | 该网络是否允许全资源池项目都可见。                    |

| 参数                        | 参数类型    | 描述                          |
|---------------------------|---------|-----------------------------|
| router:external           | boolean | 该网络是否为外部网络。                 |
| availability_zone_hints   | array   | 该网络的可用区候补选项。                |
| availability_zones        | array   | 该网络的可用区。                    |
| subnets                   | array   | 该网络的状态。                     |
| provider:network_type     | string  | 该网络的类型。例如flat、vlan、geneve。  |
| provider:physical_network | string  | 该网络的segment来自哪个物理网络。        |
| provider:segmentation_id  | string  | 该网络的segmentation ID。        |
| ipv4_address_scope        | string  | 该网络关联的ipv4地址scope。          |
| status                    | string  | 该网络的状态。                     |
| description               | string  | 该网络的自定义描述。                  |
| tags                      | array   | 该网络的自定义标签。                  |
| updated_at                | string  | 该网络最新更新时间。                  |
| created_at                | string  | 该网络创建时间。                    |
| qos_policy_id             | string  | 该网络绑定的qos策略ID。              |
| revision_number           | int     | 该网络的操作码。                    |
| ipv6_address_scope        | string  | 该网络关联的ipv6地址scope。          |
| igmp_snooping             | dict    | 该网络的二层组播配置。enabled：是否开启二层组播 |

## 请求示例

```
GET http://{endpoint}/v2.0/networks
```

## 正常响应示例

```
{
  "network": {
    "id": "553f02e3-b0d0-4f9a-a818-42e0c82448ba",
    "name": "test",
    "tenant_id": "5608307e73d845918339684a78503e78",
    "project_id": "5608307e73d845918339684a78503e78",
    "admin_state_up": true,
    "port_security_enabled": true,
    "mtu": 1500,
    "shared": false,
    "router:external": false,
    "availability_zone_hints": ["default-az"],
    "availability_zones": ["default-az"],
    "subnets": [
      "0ce8208f-8c50-4a10-92e6-894edc025e2a",
      "a6b9609c-aac5-4cf0-9IDc0c-45b247de6228"
    ],
    "igmp_snooping": {
      "enabled": true
    },
    "provider:network_type": "vlan",
    "provider:physical_network": "physnet2",
    "provider:segmentation_id": 1015,
    "ipv4_address_scope": null,
    "status": "ACTIVE",
    "description": "",
    "tags": [ ],
    "updated_at": "2021-02-19T03:54:41Z",
    "created_at": "2021-02-19T03:04:35Z",
    "qos_policy_id": null,
    "revision_number": 7,
    "ipv6_address_scope": null
  }
}
```

## 正常响应代码

200

## 错误码

401

## 创建网络

### 功能介绍

创建一个网络。

### URI

POST /v2.0/networks

### 请求消息

| 参数                        | 参数类型    | 是否必选 | Description                |
|---------------------------|---------|------|----------------------------|
| name                      | string  | 是    | 该网络的名称。默认是一个空字符串。          |
| project_id                | string  | 否    | 该网络所属的项目ID。                |
| availability_zone_hints   | array   | 否    | 该网络所属的可用区候选。               |
| availability_zones        | array   | 否    | 该网络所属的可用区。                 |
| shared                    | boolean | 否    | 该网络是否全局共享。                 |
| router:external           | boolean | 否    | 该网络是否为外部网络。默认是内部网络。        |
| provider:network_type     | string  | 否    | 该网络的类型。例如flat、vlan、geneve。 |
| provider:physical_network | string  | 否    | 该网络的segment来自哪个物理网络。       |

| 参数                       | 参数类型  | 是否必选 | Description                  |
|--------------------------|-------|------|------------------------------|
| provider:segmentation_id | int   | 否    | 该网络的segmentation ID。         |
| subnets                  | array | 否    | 该网络中的子网。                     |
| tags                     | array | 否    | 该网络的标签。                      |
| igmp_snooping            | dict  | 否    | 该网络的二层组播配置。enabled: 是否开启二层组播 |

## 响应消息

| 参数                      | 参数类型    | 描述                                   |
|-------------------------|---------|--------------------------------------|
| id                      | string  | 网络ID。                                |
| tenant_id               | string  | 项目ID。                                |
| project_id              | string  | 项目ID。                                |
| name                    | string  | 资源的人类可读名称。                           |
| admin_state_up          | boolean | 该网络是否启动。                             |
| port_security_enabled   | boolean | 该网络是否开启网络安全防护功能。如果使用安全组功能，此配置项一定为打开。 |
| mtu                     | int     | 该网络数据包最大传输单元大小。                      |
| shared                  | boolean | 该网络是否允许全资源池项目都可见。                    |
| router:external         | boolean | 该网络是否为外部网络。                          |
| availability_zone_hints | array   | 该网络的可用区候补选项。                         |
| availability_zones      | array   | 该网络的可用区。                             |
| subnets                 | array   | 该网络的状态。                              |

| 参数                        | 参数类型   | 描述                          |
|---------------------------|--------|-----------------------------|
| provider:network_type     | string | 该网络的类型。例如flat、vlan、geneve。  |
| provider:physical_network | string | 该网络的segment来自哪个物理网络。        |
| provider:segmentation_id  | string | 该网络的segmentation ID。        |
| ipv4_address_scope        | string | 该网络关联的ipv4地址scope。          |
| status                    | string | 该网络的状态。                     |
| description               | string | 该网络的自定义描述。                  |
| tags                      | array  | 该网络的自定义标签。                  |
| updated_at                | string | 该网络最新更新时间。                  |
| created_at                | string | 该网络创建时间。                    |
| qos_policy_id             | string | 该网络绑定的qos策略ID。              |
| revision_number           | int    | 该网络的操作码。                    |
| ipv6_address_scope        | string | 该网络关联的ipv6地址scope。          |
| igmp_snooping             | dict   | 该网络的二层组播配置。enabled：是否开启二层组播 |

## 请求示例

```
POST http://{endpoint}/v2.0/networks
```

```
{
  "network": {
    "name": "test"
  }
}
```

```
}  
}
```

## 正常响应示例

```
{  
  "network": {  
    "id": "553f02e3-b0d0-4f9a-a818-42e0c82448ba",  
    "name": "test",  
    "tenant_id": "5608307e73d845918339684a78503e78",  
    "project_id": "5608307e73d845918339684a78503e78",  
    "admin_state_up": true,  
    "port_security_enabled": true,  
    "mtu": 1500,  
    "shared": false,  
    "igmp_snooping": {  
      "enabled": false  
    },  
    "router:external": false,  
    "availability_zone_hints": ["default-az"],  
    "availability_zones": ["default-az"],  
    "subnets": [],  
    "provider:network_type": "vlan",  
    "provider:physical_network": "physnet2",  
    "provider:segmentation_id": 1015,  
    "ipv4_address_scope": null,  
    "status": "ACTIVE",  
    "description": "",  
    "tags": [],  
    "updated_at": "2021-02-19T03:54:41Z",  
    "created_at": "2021-02-19T03:04:35Z",  
    "qos_policy_id": null,  
    "revision_number": 7,  
    "ipv6_address_scope": null  
  }  
}
```

## 正常响应代码

201



## 错误码

400, 401

## 显示网络详情

### 功能介绍

显示某个网络的详细信息。

### URI

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

| 参数         | 参数类型   | 是否必选 | Description      |
|------------|--------|------|------------------|
| network_id | string | 是    | 获取详情的network的ID。 |

### 请求消息

无

### 响应消息

| 参数                    | 参数类型    | 描述                                   |
|-----------------------|---------|--------------------------------------|
| id                    | string  | 网络ID。                                |
| tenant_id             | string  | 项目ID。                                |
| project_id            | string  | 项目ID。                                |
| name                  | string  | 资源的人类可读名称。                           |
| admin_state_up        | boolean | 该网络是否启动。                             |
| port_security_enabled | boolean | 该网络是否开启网络安全防护功能。如果使用安全组功能，此配置项一定为打开。 |
| mtu                   | int     | 该网络数据包最大传输单元大小。                      |

| 参数                        | 参数类型    | 描述                          |
|---------------------------|---------|-----------------------------|
| shared                    | boolean | 该网络是否允许全资源池项目都可见。           |
| router:external           | boolean | 该网络是否为外部网络。                 |
| availability_zone_hints   | array   | 该网络的可用区候补选项。                |
| availability_zones        | array   | 该网络的可用区。                    |
| subnets                   | array   | 该网络的状态。                     |
| provider:network_type     | string  | 该网络的类型。例如flat、vlan、geneve。  |
| provider:physical_network | string  | 该网络的segment来自哪个物理网络。        |
| provider:segmentation_id  | string  | 该网络的segmentation ID。        |
| ipv4_address_scope        | string  | 该网络关联的ipv4地址scope。          |
| status                    | string  | 该网络的状态。                     |
| description               | string  | 该网络的自定义描述。                  |
| tags                      | array   | 该网络的自定义标签。                  |
| updated_at                | string  | 该网络最新更新时间。                  |
| created_at                | string  | 该网络创建时间。                    |
| qos_policy_id             | string  | 该网络绑定的qos策略ID。              |
| revision_number           | int     | 该网络的操作码。                    |
| ipv6_address_scope        | string  | 该网络关联的ipv6地址scope。          |
| igmp_snooping             | dict    | 该网络的二层组播配置。enabled：是否开启二层组播 |

## 请求示例

```
GET http://{endpoint}/v2.0/networks/553f02e3-b0d0-4f9a-a818-42e0c82448ba
```

## 正常响应示例

```
{
  "network": {
    "id": "553f02e3-b0d0-4f9a-a818-42e0c82448ba",
    "name": "test",
    "tenant_id": "5608307e73d845918339684a78503e78",
    "project_id": "5608307e73d845918339684a78503e78",
    "admin_state_up": true,
    "port_security_enabled": true,
    "mtu": 1500,
    "shared": false,
    "igmp_snooping": {
      "enabled": true
    },
    "router:external": false,
    "availability_zone_hints": ["default-az"],
    "availability_zones": ["default-az"],
    "subnets": [],
    "provider:network_type": "vlan",
    "provider:physical_network": "physnet2",
    "provider:segmentation_id": 1015,
    "ipv4_address_scope": null,
    "status": "ACTIVE",
    "description": "",
    "tags": [],
    "updated_at": "2021-02-19T03:54:41Z",
    "created_at": "2021-02-19T03:04:35Z",
    "qos_policy_id": null,
    "revision_number": 7,
    "ipv6_address_scope": null
  }
}
```

## 正常响应代码

200

## 错误码

400, 401

## 更新网络

### 功能介绍

更新一个网络。

### URI

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

| 参数         | 参数类型   | 是否必选 | Description      |
|------------|--------|------|------------------|
| network_id | string | 是    | 需要更新的network的ID。 |

### 请求消息

| 参数            | 参数类型   | 是否必选 | Description                 |
|---------------|--------|------|-----------------------------|
| name          | string | 否    | 网络更新后的name。                 |
| igmp_snooping | dict   | 否    | 该网络的二层组播配置。enabled：是否开启二层组播 |

### 响应消息

| 参数        | 参数类型   | 描述    |
|-----------|--------|-------|
| id        | string | 网络ID。 |
| tenant_id | string | 项目ID。 |

| 参数                        | 参数类型    | 描述                                   |
|---------------------------|---------|--------------------------------------|
| project_id                | string  | 项目ID。                                |
| name                      | string  | 资源的人类可读名称。                           |
| admin_state_up            | boolean | 该网络是否启动。                             |
| port_security_enabled     | boolean | 该网络是否开启网络安全防护功能。如果使用安全组功能，此配置项一定为打开。 |
| mtu                       | int     | 该网络数据包最大传输单元大小。                      |
| shared                    | boolean | 该网络是否允许全资源池项目都可见。                    |
| router:external           | boolean | 该网络是否为外部网络。                          |
| availability_zone_hints   | array   | 该网络的可用区候补选项。                         |
| availability_zones        | array   | 该网络的可用区。                             |
| subnets                   | array   | 该网络的状态。                              |
| provider:network_type     | string  | 该网络的类型。例如flat、vlan、geneve。           |
| provider:physical_network | string  | 该网络的segment来自哪个物理网络。                 |
| provider:segmentation_id  | string  | 该网络的segmentation ID。                 |
| ipv4_address_scope        | string  | 该网络关联的ipv4地址scope。                   |
| status                    | string  | 该网络的状态。                              |
| description               | string  | 该网络的自定义描述。                           |
| tags                      | array   | 该网络的自定义标签。                           |
| updated_at                | string  | 该网络最新更新时间。                           |

| 参数                 | 参数类型   | 描述                          |
|--------------------|--------|-----------------------------|
| created_at         | string | 该网络创建时间。                    |
| qos_policy_id      | string | 该网络绑定的qos策略ID。              |
| revision_number    | int    | 该网络的操作码。                    |
| ipv6_address_scope | string | 该网络关联的ipv6地址scope。          |
| igmp_snooping      | dict   | 该网络的二层组播配置。enabled：是否开启二层组播 |

## 请求示例

```
PUT http://{endpoint}/v2.0/networks/553f02e3-b0d0-4f9a-a818-42e0c82448ba
```

```
{
  "network": {
    "name": "test-new",
    "igmp_snooping": {
      "enabled": true
    }
  }
}
```

## 正常响应示例

```
{
  "network": {
    "id": "553f02e3-b0d0-4f9a-a818-42e0c82448ba",
    "name": "test-new",
    "tenant_id": "5608307e73d845918339684a78503e78",
    "project_id": "5608307e73d845918339684a78503e78",
    "admin_state_up": true,
    "port_security_enabled": true,
    "mtu": 1500,
    "shared": false,
```

```
    "igmp_snooping": {
      "enabled": true
    },
    "router:external": false,
    "availability_zone_hints": ["default-az"],
    "availability_zones": ["default-az"],
    "subnets": [],
    "provider:network_type": "vlan",
    "provider:physical_network": "physnet2",
    "provider:segmentation_id": 1015,
    "ipv4_address_scope": null,
    "status": "ACTIVE",
    "description": "",
    "tags": [],
    "updated_at": "2021-02-19T03:54:41Z",
    "created_at": "2021-02-19T03:04:35Z",
    "qos_policy_id": null,
    "revision_number": 7,
    "ipv6_address_scope": null
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 412

## 删除网络

### 功能介绍

删除一个网络。

### URI

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

| 参数         | 参数类型   | 是否必选 | Description |
|------------|--------|------|-------------|
| network_id | string | 是    | 需要删除网络的ID。  |

## 请求消息

无

## 响应消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/networks/553f02e3-b0d0-4f9a-a818-42e0c82448ba
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 404, 409, 412

## 获取网络可用IP详情

## 功能介绍

获取一个网络以及网络中所有子网的可用IP详情。

## URI

```
GET /v2.0/network-ip-availabilities/{network_id}
```

| 参数 | 参数类型 | 是否必选 | Description |
|----|------|------|-------------|
|----|------|------|-------------|



| 参数         | 参数类型   | 是否必选 | Description          |
|------------|--------|------|----------------------|
| network_id | string | 是    | 获取可用IP情况的network的ID。 |

## 请求消息

无

## 响应消息

| 参数                     | 参数类型   | 描述               |
|------------------------|--------|------------------|
| network_id             | string | 网络ID。            |
| tenant_id              | string | 项目ID。            |
| project_id             | string | 项目ID。            |
| network_name           | string | 资源的人类可读名称。       |
| subnet_ip_availability | array  | 该网络中所有子网的可用IP详情。 |
| used_ips               | int    | 该网络已经使用的IP数量。    |
| total_ips              | int    | 该网络可用的IP数量。      |

## 请求示例

```
GET http://{endpoint}/v2.0/network-ip-availabilities/e9319d35-2fc0-4203-b6f3-bf0295cdc74b
```

## 正常响应示例

```
{
  "network_ip_availability": {
    "network_id": "e9319d35-2fc0-4203-b6f3-bf0295cdc74b",
    "network_name": "public_net",
    "tenant_id": "209b8b929715438695771f58a8a307ef",
```

```
"project_id": "209b8b929715438695771f58a8a307ef",
"subnet_ip_availability": [{
  "subnet_id": "b66b5c1f-e5cf-4566-943e-4dd0da06f786",
  "ip_version": 4,
  "cidr": "172.16.10.0/24",
  "subnet_name": "public_net__subnet",
  "used_ips": 4,
  "total_ips": 243,
  "used_ip_addresses": ["172.16.10.100",
"172.16.10.28", "172.16.10.41", "172.16.10.62"]
}, {
  "subnet_id": "cf2a5f99-fb81-430c-ad2d-6e5858027adf",
  "ip_version": 6,
  "cidr": "2233::/64",
  "subnet_name": "public_net_default_ipv6_subnet",
  "used_ips": 2,
  "total_ips": 18446744073709551614,
  "used_ip_addresses": ["2233::25f", "2233::31f"]
}],
"used_ips": 6,
"total_ips": 18446744073709551857
}
```

## 正常响应代码

200

## 错误码

400, 401

## 1.4 子网

### 子网

#### 列举子网

##### 功能介绍

列出租户有权访问的子网。

##### URI

```
GET /v2.0/subnets
```

##### 请求消息

无

##### 响应消息

| 参数              | 参数类型    | 描述               |
|-----------------|---------|------------------|
| subnets         | array   | 子网对象的列表。         |
| id              | string  | 子网ID。            |
| tenant_id       | string  | 项目ID。            |
| project_id      | string  | 项目ID。            |
| name            | string  | 资源的人类可读名称。       |
| enable_dhcp     | boolean | 子网是否开启或关闭dhcp功能。 |
| network_id      | string  | 子网所属的网络ID。       |
| dns_nameservers | array   | 子网关联的dns服务器列表。   |

| 参数                | 参数类型    | 描述                                                        |
|-------------------|---------|-----------------------------------------------------------|
| allocation_pools  | array   | 该子网的起始IP地址和结束IP地址分配池。                                     |
| host_routes       | array   | 子网的其他路由。带有目的地和下一跳参数的字典列表。                                 |
| ip_version        | integer | IP协议版本。取值为4 或 6。                                          |
| gateway_ip        | string  | 子网网关IP地址。如果该值为null，则表示该子网没有关联网关。                          |
| cidr              | string  | 子网的CIDR。                                                  |
| created_at        | string  | 创建资源的时间(UTC ISO8601格式)。                                   |
| description       | string  | 人类可读的资源描述。                                                |
| ipv6_address_mode | string  | IPv6地址分配用于指定IP地址分配的方式。取值为dhcpv6-stateful或null。            |
| ipv6_ra_mode      | string  | IPv6路由器指定网络服务是否应该为一个子网发送ICMPv6报文。取值为dhcpv6-stateful或null。 |
| revision_number   | integer | 资源的修订号。                                                   |
| service_types     | array   | 子网关联的业务类型。                                                |
| subnetpool_id     | string  | 子网关联的子网池ID。                                               |
| updated_at        | string  | 更新资源的时间(UTC ISO8601格式)。                                   |

## 请求示例

```
GET http://{endpoint}/v2.0/subnets
```

## 正常响应示例

```
{
  "subnets": [
```

```
{
  "service_types": [
  ],
  "description": "",
  "enable_dhcp": true,
  "network_id": "553f02e3-b0d0-4f9a-a818-42e0c82448ba",
  "tenant_id": "5608307e73d845918339684a78503e78",
  "created_at": "2021-02-19T03:54:41Z",
  "dns_nameservers": [
  ],
  "updated_at": "2021-02-19T03:54:41Z",
  "gateway_ip": "192.168.1.1",
  "ipv6_ra_mode": null,
  "allocation_pools": [
    {
      "start": "192.168.1.2",
      "end": "192.168.1.254"
    }
  ],
  "host_routes": [
  ],
  "revision_number": 2,
  "ip_version": 4,
  "ipv6_address_mode": null,
  "cidr": "192.168.1.0/24",
  "project_id": "5608307e73d845918339684a78503e78",
  "id": "0ce8208f-8c50-4a10-92e6-894edc025e2a",
  "subnetpool_id": null,
  "name": "my_subnet_2"
},
{
  "service_types": [
  ],
  "description": "",
  "enable_dhcp": true,
  "network_id": "132fbb7d-a615-48c3-aa3f-7fc82970e2fb",
  "tenant_id": "fe0215103828465d925e0a4c2c81bfff1",
```

```
    "created_at": "2021-02-19T10:34:42Z",
    "dns_nameservers": [

    ],
    "updated_at": "2021-02-19T10:34:43Z",
    "gateway_ip": "10.100.0.225",
    "ipv6_ra_mode": null,
    "allocation_pools": [
        {
            "start": "10.100.0.226",
            "end": "10.100.0.238"
        }
    ],
    "host_routes": [

    ],
    "revision_number": 2,
    "ip_version": 4,
    "ipv6_address_mode": null,
    "cidr": "10.100.0.224/28",
    "project_id": "fe0215103828465d925e0a4c2c81bfff1",
    "id": "0fd87cad-88ac-442c-af65-882a38b41c15",
    "subnetpool_id": null,
    "name": "my_subnet"
  }
]
```

## 正常响应代码

200

## 错误码

401

## 创建子网

### 功能介绍

在网络中创建子网。

## URI

POST /v2.0/subnets

## 请求消息

| 参数               | 参数类型    | 是否必选 | Description                                                                          |
|------------------|---------|------|--------------------------------------------------------------------------------------|
| subnet           | object  | 是    | 一个子网对象。                                                                              |
| tenant_id        | string  | 否    | 资源所属项目的ID。只有管理员和具有advsvc角色的用户可以指定他们自己以外的项目ID。不能通过授权策略更改此值。                           |
| name             | string  | 否    | 资源的人类可读名称。默认是一个空字符串。                                                                 |
| enable_dhcp      | boolean | 否    | 子网是否开启或关闭dhcp功能。默认是true。                                                             |
| network_id       | string  | 是    | 子网所属的网络ID。                                                                           |
| allocation_pools | array   | 否    | 该子网的起始IP地址和结束IP地址池。如果不配置“allocation_pools”参数，将自动把CIDR中的所有IP地址分配到IP地址池，默认不包括子网网关预留地址。 |
| host_routes      | array   | 否    | 子网的其他路由。带有目的地和下一跳参数的字典列表。默认值为空列表。                                                    |
| ip_version       | integer | 是    | IP协议版本。取值为4或6。                                                                       |
| gateway_ip       | string  | 否    | 子网网关IP地址。如果该值为null，则表示该子网没有关联网关。如果不指定gateway_ip，则会默认从CIDR中为子网分配网关地址。                 |
| cidr             | string  | 是    | 子网的CIDR。                                                                             |
| description      | string  | 否    | 人类可读的资源描述。默认是一个空字符串。                                                                 |

| 参数                | 参数类型   | 是否必选 | Description                                               |
|-------------------|--------|------|-----------------------------------------------------------|
| ipv6_address_mode | string | 否    | IPv6地址分配用于指定IP地址分配的方式。取值为dhcpv6-stateful或null。            |
| ipv6_ra_mode      | string | 否    | IPv6路由器指定网络服务是否应该为一个子网发送ICMPv6报文。取值为dhcpv6-stateful或null。 |

## 响应消息

| 参数               | 参数类型    | 描述                               |
|------------------|---------|----------------------------------|
| subnet           | object  | 一个子网对象。                          |
| id               | string  | 子网ID。                            |
| tenant_id        | string  | 项目ID。                            |
| project_id       | string  | 项目ID。                            |
| name             | string  | 资源的人类可读名称。                       |
| enable_dhcp      | boolean | 子网是否开启或关闭dhcp功能。                 |
| network_id       | string  | 子网所属的网络ID。                       |
| dns_nameservers  | array   | 子网关联的dns服务器列表。                   |
| allocation_pools | array   | 该子网的起始IP地址和结束IP地址池。              |
| host_routes      | array   | 子网的其他路由。带有目的地和下一跳参数的字典列表。        |
| ip_version       | integer | IP协议版本。取值为4 或 6。                 |
| gateway_ip       | string  | 子网网关IP地址。如果该值为null，则表示该子网没有关联网关。 |
| cidr             | string  | 子网的CIDR。                         |



| 参数                | 参数类型    | 描述                                                        |
|-------------------|---------|-----------------------------------------------------------|
| created_at        | string  | 创建资源的时间(UTC ISO8601格式)。                                   |
| description       | string  | 人类可读的资源描述。                                                |
| ipv6_address_mode | string  | IPv6地址分配方式用于指定IP地址的分配方式。取值为dhcpv6-stateful或null。          |
| ipv6_ra_mode      | string  | IPv6路由器指定网络服务是否应该为一个子网发送ICMPv6报文。取值为dhcpv6-stateful或null。 |
| revision_number   | integer | 资源的修订号。                                                   |
| service_types     | array   | 子网关联的业务类型。                                                |
| subnetpool_id     | string  | 子网关联的子网池ID。                                               |
| updated_at        | string  | 更新资源的时间(UTC ISO8601格式)。                                   |

## 请求示例

```
POST http://{endpoint}/v2.0/subnets
```

```
{
  "subnet": {
    "network_id": "605a9cd6-678f-44ed-9adc-5ad5eeeb9806",
    "ip_version": 4,
    "cidr": "192.168.4.0/24"
  }
}
```

## 正常响应示例

```
{
  "subnet": {
    "service_types": [
```

```
],
  "description": "",
  "enable_dhcp": true,
  "network_id": "605a9cd6-678f-44ed-9adc-5ad5eeeb9806",
  "tenant_id": "870c5414baee43a4ba7adfb253e46018",
  "created_at": "2021-02-24T02:24:55Z",
  "dns_nameservers": [

  ],
  "updated_at": "2021-02-24T02:24:55Z",
  "gateway_ip": "192.168.4.1",
  "ipv6_ra_mode": null,
  "allocation_pools": [
    {
      "start": "192.168.4.2",
      "end": "192.168.4.254"
    }
  ],
  "host_routes": [

  ],
  "revision_number": 2,
  "ip_version": 4,
  "ipv6_address_mode": null,
  "cidr": "192.168.4.0/24",
  "project_id": "43d9b6c8ee91457cb2e9bb86f701b3ef",
  "id": "8adac510-3352-43c2-bdca-dd017bf092ae",
  "subnetpool_id": null,
  "name": "my_subnet"
}
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409

## 显示子网详情

## 功能介绍

显示子网的详细信息。

## URI

```
GET /v2.0/subnets/{subnet_id}
```

| 参数        | 参数类型   | 是否必选 | Description |
|-----------|--------|------|-------------|
| subnet_id | string | 是    | 子网ID。       |

## 请求消息

无

## 响应消息

| 参数               | 参数类型    | 描述                      |
|------------------|---------|-------------------------|
| subnet           | object  | 一个子网对象。                 |
| id               | string  | 子网ID。                   |
| tenant_id        | string  | 项目ID。                   |
| project_id       | string  | 项目ID。                   |
| created_at       | string  | 创建资源的时间(UTC ISO8601格式)。 |
| name             | string  | 资源的人类可读名称。              |
| enable_dhcp      | boolean | 子网是否开启或关闭dhcp功能。        |
| network_id       | string  | 子网所属的网络ID。              |
| dns_nameservers  | array   | 子网关联的dns服务器列表。          |
| allocation_pools | array   | 该子网的起始IP地址和结束IP地址池。     |

| 参数                | 参数类型    | 描述                                                        |
|-------------------|---------|-----------------------------------------------------------|
| host_routes       | array   | 子网的其他路由。带有目的地和下一跳参数的字典列表。                                 |
| ip_version        | integer | IP协议版本。取值为4 或6。                                           |
| gateway_ip        | string  | 子网网关IP地址。如果该值为null，则表示该子网没有关联网关。                          |
| cidr              | string  | 子网的CIDR。                                                  |
| updated_at        | string  | 更新资源的时间(UTC ISO8601格式)。                                   |
| description       | string  | 人类可读的资源描述。                                                |
| ipv6_address_mode | string  | IPv6地址分配用于指定IP地址分配的方式。取值为dhcpv6-stateful或null。            |
| ipv6_ra_mode      | string  | IPv6路由器指定网络服务是否应该为一个子网发送ICMPv6报文。取值为dhcpv6-stateful或null。 |
| revision_number   | integer | 资源的修订号。                                                   |
| service_types     | array   | 子网关联的业务类型。                                                |
| subnetpool_id     | string  | 子网关联的子网池ID。                                               |

## 请求示例

```
GET http://{endpoint}/v2.0/subnets/008d3711-7978-439f-a340-707cb091aa85
```

## 正常响应示例

```
{
  "subnet":{
    "service_types":[

    ],
    "description":""
  },
}
```

```
"enable_dhcp":true,
"network_id":"6c7ee1a2-25ad-4b83-98e6-f9412f679576",
"tenant_id":"870c5414baee43a4ba7adfb253e46018",
"created_at":"2021-02-24T02:34:49Z",
"dns_nameservers":[

],
"updated_at":"2021-02-24T02:34:49Z",
"gateway_ip":"192.168.1.1",
"ipv6_ra_mode":null,
"allocation_pools":[
  {
    "start":"192.168.1.2",
    "end":"192.168.1.254"
  }
],
"host_routes":[

],
"revision_number":2,
"ip_version":4,
"ipv6_address_mode":null,
"cidr":"192.168.1.0/24",
"project_id":"870c5414baee43a4ba7adfb253e46018",
"id":"008d3711-7978-439f-a340-707cb091aa85",
"subnetpool_id":null,
"name":"my_subnet"
}
}
```

## 正常响应代码

200

## 错误码

400, 401

## 更新子网

## 功能介绍

更新一个子网。某些属性，如IP版本（ip\_version）和CIDR（cidr）无法更新。尝试更新这些属性会导致“400 错误请求”错误。

## URI

```
PUT /v2.0/subnets/{subnet_id}
```

| 参数        | 参数类型   | 是否必选 | Description |
|-----------|--------|------|-------------|
| subnet_id | string | 否    | 子网ID。       |

## 请求消息

| 参数               | 参数类型    | 是否必选 | Description                                                                          |
|------------------|---------|------|--------------------------------------------------------------------------------------|
| name             | string  | 否    | 资源的人类可读名称。                                                                           |
| enable_dhcp      | boolean | 否    | 子网是否开启或关闭dhcp功能。默认是true。                                                             |
| dns_nameservers  | array   | 否    | 子网关联的dns服务器列表。默认为空列表。                                                                |
| allocation_pools | array   | 否    | 该子网的起始IP地址和结束IP地址池。如果不配置“allocation_pools”参数，将自动把CIDR中的所有IP地址分配到IP地址池，默认不包括子网网关预留地址。 |
| host_routes      | array   | 否    | 子网的其他路由。带有目的地和下一跳参数的字典列表。默认值为空列表。                                                    |
| gateway_ip       | string  | 否    | 子网网关IP地址。如果该值为null，则表示该子网没有关联网关。如果不指定gateway_ip，则默认从CIDR中为子网分配网关地址。                  |

## 响应消息

| 参数                | 参数类型    | 描述                                                        |
|-------------------|---------|-----------------------------------------------------------|
| subnet            | object  | 一个子网对象。                                                   |
| id                | string  | 子网ID。                                                     |
| tenant_id         | string  | 项目ID。                                                     |
| project_id        | string  | 项目ID。                                                     |
| name              | string  | 资源的人类可读名称。                                                |
| enable_dhcp       | boolean | 子网是否开启或关闭dhcp功能。                                          |
| network_id        | string  | 子网所属的网络ID。                                                |
| dns_nameservers   | array   | 子网关联的dns服务器列表。                                            |
| allocation_pools  | array   | 该子网的起始IP地址和结束IP地址分配池。                                     |
| host_routes       | array   | 子网的其他路由。带有目的地和下一跳参数的字典列表。                                 |
| ip_version        | integer | IP协议版本。取值为4或6。                                            |
| gateway_ip        | string  | 子网网关IP地址。如果该值为null，则表示该子网没有关联网关。                          |
| cidr              | string  | 子网的CIDR。                                                  |
| created_at        | string  | 创建资源的时间(UTC ISO8601格式)。                                   |
| description       | string  | 人类可读的资源描述。                                                |
| ipv6_address_mode | string  | IPv6地址分配用于指定IP地址分配的方式。取值为dhcpv6-stateful或null。            |
| ipv6_ra_mode      | string  | IPv6路由器指定网络服务是否应该为一个子网发送ICMPv6报文。取值为dhcpv6-stateful或null。 |
| revision_number   | integer | 资源的修订号。                                                   |

| 参数            | 参数类型   | 描述                      |
|---------------|--------|-------------------------|
| service_types | array  | 子网关联的业务类型。              |
| subnetpool_id | string | 子网关联的子网池ID。             |
| updated_at    | string | 更新资源的时间(UTC ISO8601格式)。 |

## 请求示例

```
PUT http://{endpoint}/v2.0/subnets/008d3711-7978-439f-a340-707cb091aa85
```

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

## 正常响应示例

```
{
  "subnet": {
    "service_types": [],
    "description": "",
    "enable_dhcp": true,
    "network_id": "6c7ee1a2-25ad-4b83-98e6-f9412f679576",
    "tenant_id": "870c5414baee43a4ba7adfb253e46018",
    "created_at": "2021-02-24T02:34:49Z",
    "dns_nameservers": [],
    "updated_at": "2021-02-24T02:59:21Z",
    "gateway_ip": "192.168.1.1",
    "ipv6_ra_mode": null,
    "allocation_pools": [{
      "start": "192.168.1.2",
      "end": "192.168.1.254"
    }],
    "host_routes": [],
    "revision_number": 3,
  }
}
```



```
"ip_version": 4,
"ipv6_address_mode": null,
"cidr": "192.168.1.0/24",
"project_id": "870c5414baee43a4ba7adfb253e46018",
"id": "008d3711-7978-439f-a340-707cb091aa85",
"subnetpool_id": null,
"name": "my_subnet"
}
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 412

## 删除子网

### 功能介绍

删除一个子网。如果子网IP仍然处于被分配状态，则操作失败。

### URI

```
DELETE /v2.0/subnets/{subnet_id}
```

| 参数        | 参数类型   | 是否必选 | Description |
|-----------|--------|------|-------------|
| subnet_id | string | 否    | 子网ID。       |

### 请求消息

无

### 响应消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/subnets/008d3711-7978-439f-a340-707cb091aa85
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 404, 412

## 1.5 虚拟网卡

### 虚拟网卡

#### 列举虚拟网卡

##### 功能介绍

列举租户有权限访问的网卡

##### URI

```
GET /v2.0/ports
```

##### 请求消息

| 参数         | 参数类型   | 是否必选 | 描述                                                                                                                           |
|------------|--------|------|------------------------------------------------------------------------------------------------------------------------------|
| fields     | string | 否    | 使用 <code>fields</code> 参数, API只返回请求的属性集,可以指定多次。例如, URL中指定' <code>fields=id&amp;fields=name</code> ', 那么API只会返回'id'和'name'属性。 |
| project_id | string | 否    | 指定项目ID进行查询。                                                                                                                  |

##### 响应消息

| 参数                      | 参数类型    | 描述                                                  |
|-------------------------|---------|-----------------------------------------------------|
| ports                   | array   | 网卡列表。                                               |
| admin_state_up          | boolean | 网卡的 admin 状态, 取值 true 或 false。                      |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表, 每个 pair 包含一个单独 IP 或一个 IP 和 MAC 。 |

| 参数                    | 参数类型    | 描述                                                                                                  |
|-----------------------|---------|-----------------------------------------------------------------------------------------------------|
| binding:host_id       | string  | 网卡所属的 host ID。                                                                                      |
| binding:profile       | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                                                        |
| binding:vif_details   | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。                                                     |
| binding:vif_type      | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs，bridge，macvtap，未绑定是 unbound，失败是 binding_failed。                             |
| binding:vnic_type     | string  | 网卡挂载的类型，比如 normal，direct，baremetal。                                                                 |
| created_at            | string  | 网卡创建时间 (in UTC ISO8601 format)。                                                                     |
| description           | string  | 用户描述。                                                                                               |
| device_id             | string  | 使用该网卡的设备 ID，比如一个虚拟机 uuid。                                                                           |
| device_owner          | string  | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port，network:vip 表示虚拟 IP，network:secondary 表示辅助私网 IP。 |
| extra_dhcp_opts       | array   | 额外的 DHCP 参数列表。                                                                                      |
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                                                        |
| id                    | string  | 网卡的 ID。                                                                                             |
| mac_address           | string  | 网卡的 MAC 地址。                                                                                         |
| name                  | string  | 网卡的名称。                                                                                              |
| network_id            | string  | 网卡所属的网络 ID。                                                                                         |
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。                                  |
| project_id            | string  | 网卡所属的 project ID。                                                                                   |

| 参数                    | 参数类型    | 描述                                    |
|-----------------------|---------|---------------------------------------|
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                      |
| revision_number       | integer | 网卡的版本号。                               |
| security_groups       | array   | 网卡关联的安全组列表。                           |
| status                | string  | 网卡的状态： ACTIVE, DOWN, BUILD and ERROR。 |
| tags                  | array   | 网卡的标签列表。                              |
| qos_network_policy_id | string  | 网卡所属 network 绑定的 Qos policy ID。       |
| tenant_id             | string  | 网卡所属的 tenant ID。                      |
| updated_at            | string  | 网卡最近更新的时间(in UTC ISO8601 format)。     |

## 请求示例

```
GET https://{endpoint}/v2.0/ports
```

## 正常响应示例

```
{
  "ports": [
    {
      "admin_state_up": true,
      "allowed_address_pairs": [],
      "binding:host_id": "node-10.domain.tld",
      "binding:profile": {},
      "binding:vif_details": {
        "ovs_hybrid_plug": false,
        "port_filter": true
      },
      "binding:vif_type": "ovs",
      "binding:vnic_type": "normal",
      "created_at": "2021-02-24T02:41:53Z",
```

```
"description": "",
"device_id": "f9583498-c576-4e2e-86ad-c6a662b000a8",
"device_owner": "compute:kuryr",
"extra_dhcp_opts": [],
"fixed_ips": [
  {
    "ip_address": "192.168.111.12",
    "subnet_id": "f32a3ab8-6a5a-41c6-a202-cfe7bea9348a"
  }
],
"id": "04c4339b-d13f-4f14-9be5-8f7477cbea4e",
"mac_address": "fa:16:3e:f8:72:bb",
"name": "d717c61a2",
"network_id": "7b0343bd-cf33-45af-bb2b-25bea12eb2c9",
"port_security_enabled": true,
"project_id": "d717c61a263f4a0b9271d0a271653c2a",
"qos_policy_id": null,
"revision_number": 8,
"security_groups": [
  "a8c01103-2cf1-4f6a-a8f1-ed84ec2f794f"
],
"status": "ACTIVE",
"qos_network_policy_id": null,
"tenant_id": "d717c61a263f4a0b9271d0a271653c2a",
"updated_at": "2021-02-24T02:42:13Z"
}
]
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 创建虚拟网卡

## 功能介绍

创建一个虚拟网卡

## URI

POST /v2.0/ports

## 请求消息

| 参数                      | 参数类型    | 是否必选 | Description                                                                                         |
|-------------------------|---------|------|-----------------------------------------------------------------------------------------------------|
| port                    | object  | 是    | 网卡对象。                                                                                               |
| admin_state_up          | boolean | 否    | 网卡的admin状态，取值true或false。                                                                            |
| allowed_addresses_pairs | array   | 否    | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC 。                                                  |
| binding:host_id         | string  | 否    | 网卡所属的 host ID。                                                                                      |
| binding:profile         | object  | 否    | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                                                        |
| binding:vnic_type       | string  | 否    | 网卡挂载的类型，比如normal，direct，baremetal，默认normal。                                                         |
| description             | string  | 否    | 用户描述。                                                                                               |
| device_id               | string  | 否    | 使用该网卡的设备ID，比如一个虚拟机uuid。                                                                             |
| device_owner            | string  | 否    | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port，network:vip 表示虚拟 IP，network:secondary 表示辅助私网 IP。 |
| extra_dhcp_options      | array   | 否    | 额外的 DHCP 参数列表。                                                                                      |
| fixed_ips               | array   | 否    | 网卡的固定 IP 列表。                                                                                        |
| mac_address             | string  | 否    | 网卡的 MAC 地址。                                                                                         |

| 参数                    | 参数类型    | 是否必选 | Description                                                  |
|-----------------------|---------|------|--------------------------------------------------------------|
| name                  | string  | 否    | 网卡的名称。                                                       |
| network_id            | string  | 是    | 网卡所属的网络ID。                                                   |
| port_security_enabled | boolean | 否    | 网卡的安全状态。true或者false，如果是true将会配置安全组规则和anti-spoofing规则，否则不会处理。 |
| project_id            | string  | 否    | 网卡所属的 project ID。                                            |
| qos_policy_id         | string  | 否    | 网卡关联的 QoS 策略 ID。                                             |
| security_groups       | array   | 否    | 网卡关联的安全组列表。                                                  |
| tags                  | array   | 否    | 网卡的标签列表。                                                     |
| tenant_id             | string  | 否    | 网卡所属的 tenant ID。                                             |

## 响应消息

| 参数                      | 参数类型    | 描述                                                                        |
|-------------------------|---------|---------------------------------------------------------------------------|
| ports                   | array   | 网卡列表。                                                                     |
| admin_state_up          | boolean | 网卡的 admin 状态，取值 true 或 false。                                             |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC。                         |
| binding:host_id         | string  | 网卡所属的 host ID。                                                            |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                              |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。                           |
| binding:vif_type        | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap，未绑定是 unbound，失败是 binding_failed。 |



| 参数                    | 参数类型    | 描述                                                                                                    |
|-----------------------|---------|-------------------------------------------------------------------------------------------------------|
| binding:vnic_type     | string  | 网卡挂载的类型，比如 normal, direct, baremetal。                                                                 |
| created_at            | string  | 网卡创建时间 (in UTC ISO8601 format)。                                                                       |
| description           | string  | 用户描述。                                                                                                 |
| device_id             | string  | 使用该网卡的设备 ID，比如一个虚机 uuid。                                                                              |
| device_owner          | string  | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port, network:vip 表示虚拟 IP, network:secondary 表示辅助私网 IP。 |
| extra_dhcp_opts       | array   | 额外的 DHCP 参数列表。                                                                                        |
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                                                          |
| id                    | string  | 网卡的 ID。                                                                                               |
| mac_address           | string  | 网卡的 MAC 地址。                                                                                           |
| name                  | string  | 网卡的名称。                                                                                                |
| network_id            | string  | 网卡所属的网络 ID。                                                                                           |
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。                                    |
| project_id            | string  | 网卡所属的 project ID。                                                                                     |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                                                      |
| revision_number       | integer | 网卡的版本号。                                                                                               |
| security_groups       | array   | 网卡关联的安全组列表。                                                                                           |
| status                | string  | 网卡的状态：ACTIVE, DOWN, BUILD and ERROR。                                                                  |
| tags                  | array   | 网卡的标签列表。                                                                                              |

| 参数                    | 参数类型   | 描述                                |
|-----------------------|--------|-----------------------------------|
| qos_network_policy_id | string | 网卡所属network绑定的 QoS policy ID。     |
| tenant_id             | string | 网卡所属的 tenant ID。                  |
| updated_at            | string | 网卡最近更新的时间(in UTC ISO8601 format)。 |

## 请求示例

```
POST http://{endpoint}/v2.0/ports
```

```
{
  "port": {
    "admin_state_up": true,
    "name": "test",
    "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2"
  }
}
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnictype": "normal",
    "created_at": "2021-02-24T07:01:14Z",
    "description": "",
    "device_id": "",
    "device_owner": "",
    "extra_dhcp_opts": [],
    "fixed_ips": [
```

```
{
  {
    "ip_address": "172.16.0.14",
    "subnet_id": "95347c6c-181d-4739-998d-0365487c9ac6"
  },
  "id": "602debb5-289d-4ecc-bc86-6a2f84a4b677",
  "mac_address": "fa:16:3e:91:6b:15",
  "name": "test",
  "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2",
  "port_security_enabled": true,
  "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
  "qos_policy_id": null,
  "revision_number": 5,
  "security_groups": [
    "65db6fc3-7ba5-42e9-95f1-d45145f1e65f"
  ],
  "status": "DOWN",
  "qos_network_policy_id": null,
  "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
  "updated_at": "2021-02-24T07:01:15Z"
}
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 创建辅助私网IP网卡

### 功能介绍

创建一个辅助私网IP的网卡。使用本辅助私网 IP，需将辅助私网 IP 地址添加到业务网卡的 `allowed_address_pairs` 里，可以参考[增加虚拟网卡的allowed\\_address\\_pairs](#)。

### URI

POST /v2.0/ports

## 请求消息

| 参数           | 参数类型   | 是否必选 | Description                     |
|--------------|--------|------|---------------------------------|
| port         | object | 是    | 网卡对象。                           |
| description  | string | 否    | 用户描述。                           |
| device_id    | string | 是    | 指定使用该辅助私网 IP 的 port ID。         |
| device_owner | string | 是    | 指定为 network:secondary表示辅助私网 IP。 |
| fixed_ips    | array  | 否    | 网卡的固定 IP 列表。                    |
| name         | string | 否    | 网卡的名称。                          |
| network_id   | string | 是    | 网卡所属的网络 ID。                     |
| tenant_id    | string | 否    | 网卡所属的 tenant ID。                |

## 响应消息

| 参数                      | 参数类型    | 描述                                                 |
|-------------------------|---------|----------------------------------------------------|
| ports                   | array   | 网卡列表。                                              |
| admin_state_up          | boolean | 网卡的 admin 状态，取值 true 或 false。                      |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC 。 |
| binding:host_id         | string  | 网卡所属的 host ID。                                     |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                       |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。    |

| 参数                    | 参数类型    | 描述                                                                          |
|-----------------------|---------|-----------------------------------------------------------------------------|
| binding:vif_type      | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap, 未绑定是 unbound, 失败是 binding_failed。 |
| binding:vnic_type     | string  | 网卡挂载的类型，比如 normal, direct, baremetal。                                       |
| created_at            | string  | 网卡创建时间 (in UTC ISO8601 format)。                                             |
| description           | string  | 用户描述。                                                                       |
| device_id             | string  | 使用该网卡的设备 ID，比如一个虚机 uuid。                                                    |
| device_owner          | string  | network:secondary 表示辅助私网 IP。                                                |
| extra_dhcp_opts       | array   | 额外的 DHCP 参数列表。                                                              |
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                                |
| id                    | string  | 网卡的 ID。                                                                     |
| mac_address           | string  | 网卡的 MAC 地址。                                                                 |
| name                  | string  | 网卡的名称。                                                                      |
| network_id            | string  | 网卡所属的网络 ID。                                                                 |
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。          |
| project_id            | string  | 网卡所属的 project ID。                                                           |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                            |
| revision_number       | integer | 网卡的版本号。                                                                     |
| security_groups       | array   | 网卡关联的安全组列表。                                                                 |
| status                | string  | 网卡的状态：ACTIVE, DOWN, BUILD and ERROR。                                        |
| tags                  | array   | 网卡的标签列表。                                                                    |

| 参数                    | 参数类型   | 描述                                |
|-----------------------|--------|-----------------------------------|
| qos_network_policy_id | string | 网卡所属network绑定的 QoS policy ID。     |
| tenant_id             | string | 网卡所属的 tenant ID。                  |
| updated_at            | string | 网卡最近更新的时间(in UTC ISO8601 format)。 |

## 请求示例

```
POST http://{endpoint}/v2.0/ports
```

```
{
  "port": {
    "name": "secondary port",
    "device_id": "34b23ff6-aae0-411a-89c8-dba87be9bf59",
    "device_owner": "network:secondary",
    "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2"
  }
}
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnic_type": "normal",
    "created_at": "2021-02-24T07:01:14Z",
    "description": "",
    "device_id": "34b23ff6-aae0-411a-89c8-dba87be9bf59",
    "device_owner": "network:secondary",
    "extra_dhcp_opts": [],

```

```
    "fixed_ips": [
      {
        "ip_address": "172.16.0.14",
        "subnet_id": "95347c6c-181d-4739-998d-0365487c9ac6"
      }
    ],
    "id": "602debb5-289d-4ecc-bc86-6a2f84a4b677",
    "mac_address": "fa:16:3e:91:6b:15",
    "name": "secondary port",
    "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2",
    "port_security_enabled": true,
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "qos_policy_id": null,
    "revision_number": 5,
    "security_groups": [
      "65db6fc3-7ba5-42e9-95f1-d45145f1e65f"
    ],
    "status": "DOWN",
    "qos_network_policy_id": null,
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "updated_at": "2021-02-24T07:01:15Z"
  }
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 创建二层桥接网卡

### 功能介绍

创建一个二层桥接类型的网卡。

### URI

POST /v2.0/ports

## 请求消息

| 参数               | 参数类型   | 是否必选 | Description                                                                                                                                                                               |
|------------------|--------|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| port             | object | 是    | 网卡对象。                                                                                                                                                                                     |
| binding:host_id  | string | 是    | 网卡所属的 host ID。                                                                                                                                                                            |
| binding:profile  | object | 是    | 网卡信息字典。需要在local_link_information内指定vlan_id和switch_info信息，其中vlan_id是该二层桥接承载的云外流量的vlan ID，switch_info是二层桥接网关节点的可用域信息。例如"local_link_information":[{"vlan_id": 100, "switch_info": "leaf1"}]。 |
| binding:vnictype | string | 是    | 网卡挂载的类型，设置为 baremetal。                                                                                                                                                                    |
| description      | string | 否    | 用户描述。                                                                                                                                                                                     |
| device_owner     | string | 是    | 指定为 network:l2_gateway 表示二层桥接网卡。                                                                                                                                                          |
| fixed_ips        | array  | 是    | 网卡的固定 IP 列表，设置为空列表。                                                                                                                                                                       |
| name             | string | 否    | 网卡的名称。                                                                                                                                                                                    |
| network_id       | string | 是    | 网卡所属的网络 ID。                                                                                                                                                                               |
| qos_policy_id    | string | 否    | 网卡关联的 QoS 策略 ID。                                                                                                                                                                          |
| tenant_id        | string | 否    | 网卡所属的 tenant ID。                                                                                                                                                                          |

## 响应消息

| 参数   | 参数类型   | 描述    |
|------|--------|-------|
| port | object | 网卡对象。 |



| 参数                      | 参数类型    | 描述                                                                         |
|-------------------------|---------|----------------------------------------------------------------------------|
| admin_state_up          | boolean | 网卡的 admin 状态，取值 true 或 false。                                              |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC。                          |
| binding:host_id         | string  | 网卡所属的 host ID。                                                             |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                               |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。                            |
| binding:vif_type        | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap, 未绑定是 unbound，失败是 binding_failed。 |
| binding:vnic_type       | string  | 网卡挂载的类型，比如 normal, direct, baremetal。                                      |
| created_at              | string  | 网卡创建时间 (in UTC ISO8601 format)。                                            |
| description             | string  | 用户描述。                                                                      |
| device_id               | string  | 使用该网卡的设备 ID，比如一个虚机 uuid。                                                   |
| device_owner            | string  | network:secondary 表示辅助私网 IP。                                               |
| extra_dhcp_opts         | array   | 额外的 DHCP 参数列表。                                                             |
| fixed_ips               | array   | 网卡的固定 IP 列表。                                                               |
| id                      | string  | 网卡的 ID。                                                                    |
| mac_address             | string  | 网卡的 MAC 地址。                                                                |
| name                    | string  | 网卡的名称。                                                                     |
| network_id              | string  | 网卡所属的网络 ID。                                                                |
| port_security_enabled   | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。         |

| 参数                    | 参数类型    | 描述                                    |
|-----------------------|---------|---------------------------------------|
| project_id            | string  | 网卡所属的 project ID。                     |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                      |
| revision_number       | integer | 网卡的版本号。                               |
| security_groups       | array   | 网卡关联的安全组列表。                           |
| status                | string  | 网卡的状态： ACTIVE, DOWN, BUILD and ERROR。 |
| tags                  | array   | 网卡的标签列表。                              |
| qos_network_policy_id | string  | 网卡所属network绑定的 QoS policy ID。         |
| tenant_id             | string  | 网卡所属的 tenant ID。                      |
| updated_at            | string  | 网卡最近更新的时间(in UTC ISO8601 format)。     |

## 请求示例

```
POST http://{endpoint}/v2.0/ports
```

```
{
  "port": {
    "binding:host_id": "l2gw-host",
    "binding:profile": {
      "local_link_information": [
        {
          "switch_info": "default-az",
          "vlan_id": 198
        }
      ]
    },
    "binding:vnic_type": "baremetal",
    "device_owner": "network:l2_gateway",
    "fixed_ips": [],
  }
}
```

```
    "name": "l2gw-port",  
    "network_id": "eeb9374d-548d-4efc-8c79-ae5890f33a47"  
  }  
}
```

## 正常响应示例

```
{  
  "port": {  
    "admin_state_up": true,  
    "allowed_address_pairs": [],  
    "binding:host_id": "l2gw-host",  
    "binding:profile": {  
      "local_link_information": [  
        {  
          "switch_info": "default-az",  
          "vlan_id": 198  
        }  
      ]  
    },  
    "binding:vif_details": {  
      "switch_info": "default-az",  
      "vlan_id": 198  
    },  
    "binding:vif_type": "other",  
    "binding:vnic_type": "baremetal",  
    "created_at": "2024-12-25T03:03:55Z",  
    "description": "",  
    "device_id": "",  
    "device_owner": "network:l2_gateway",  
    "extra_dhcp_opts": [],  
    "fixed_ips": [],  
    "id": "cda3f270-d4a1-469d-bcd3-d26aae5c80e7",  
    "mac_address": "fa:16:3e:df:e4:7a",  
    "name": "l2gw-port",  
    "network_id": "eeb9374d-548d-4efc-8c79-ae5890f33a47",  
    "port_security_enabled": false,  
    "project_id": "ed6255c8005846889398598becb9878f",  
    "qos_network_policy_id": null,  
    "qos_policy_id": null,  
  }  
}
```

```
{
  "revision_number": 2,
  "security_groups": [],
  "status": "DOWN",
  "tags": [],
  "tenant_id": "ed6255c8005846889398598becb9878f",
  "updated_at": "2024-12-25T03:03:55Z"
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 创建虚拟IP网卡

### 功能介绍

创建一个虚拟IP的网卡。使用本虚拟IP，需将虚拟IP地址添加到业务网卡的 `allowed_address_pairs` 里，可以参考[增加虚拟网卡的allowed\\_address\\_pairs](#)。

### URI

POST /v2.0/ports

### 请求消息

| 参数           | 参数类型   | 是否必选 | Description             |
|--------------|--------|------|-------------------------|
| port         | object | 是    | 网卡对象。                   |
| description  | string | 否    | 用户描述。                   |
| device_owner | string | 是    | 指定为 network:vip 表示虚拟IP。 |
| fixed_ips    | array  | 否    | 网卡的固定 IP 列表。            |
| name         | string | 否    | 网卡的名称。                  |

| 参数         | 参数类型   | 是否必选 | Description      |
|------------|--------|------|------------------|
| network_id | string | 是    | 网卡所属的网络 ID。      |
| tenant_id  | string | 否    | 网卡所属的 tenant ID。 |

## 响应消息

| 参数                      | 参数类型    | 描述                                                                         |
|-------------------------|---------|----------------------------------------------------------------------------|
| ports                   | array   | 网卡列表。                                                                      |
| admin_state_up          | boolean | 网卡的 admin 状态，取值 true 或 false。                                              |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC。                          |
| binding:host_id         | string  | 网卡所属的 host ID。                                                             |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                               |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。                            |
| binding:vif_type        | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap, 未绑定是 unbound，失败是 binding_failed。 |
| binding:vnictype        | string  | 网卡挂载的类型，比如 normal, direct, baremetal。                                      |
| created_at              | string  | 网卡创建时间 (in UTC ISO8601 format)。                                            |
| description             | string  | 用户描述。                                                                      |
| device_id               | string  | 使用该网卡的设备 ID，比如一个虚拟机 uuid。                                                  |
| device_owner            | string  | network:vip 表示虚拟 IP。                                                       |
| extra_dhcp_options      | array   | 额外的 DHCP 参数列表。                                                             |

| 参数                    | 参数类型    | 描述                                                                 |
|-----------------------|---------|--------------------------------------------------------------------|
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                       |
| id                    | string  | 网卡的 ID。                                                            |
| mac_address           | string  | 网卡的 MAC 地址。                                                        |
| name                  | string  | 网卡的名称。                                                             |
| network_id            | string  | 网卡所属的网络 ID。                                                        |
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。 |
| project_id            | string  | 网卡所属的 project ID。                                                  |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                   |
| revision_number       | integer | 网卡的版本号。                                                            |
| security_groups       | array   | 网卡关联的安全组列表。                                                        |
| status                | string  | 网卡的状态：ACTIVE, DOWN, BUILD and ERROR。                               |
| tags                  | array   | 网卡的标签列表。                                                           |
| qos_network_policy_id | string  | 网卡所属network绑定的 QoS policy ID。                                      |
| tenant_id             | string  | 网卡所属的 tenant ID。                                                   |
| updated_at            | string  | 网卡最近更新的时间(in UTC ISO8601 format)。                                  |

## 请求示例

```
POST http://{endpoint}/v2.0/ports
```

```
{
  "port": {
    "name": "VIP port",
```

```
    "device_owner": "network:vip",
    "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2"
  }
}
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnictype": "normal",
    "created_at": "2021-02-24T07:01:14Z",
    "description": "",
    "device_id": "",
    "device_owner": "network:vip",
    "extra_dhcp_opts": [],
    "fixed_ips": [
      {
        "ip_address": "172.16.0.14",
        "subnet_id": "95347c6c-181d-4739-998d-0365487c9ac6"
      }
    ],
    "id": "602debb5-289d-4ecc-bc86-6a2f84a4b677",
    "mac_address": "fa:16:3e:91:6b:15",
    "name": "VIP port",
    "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2",
    "port_security_enabled": true,
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "qos_policy_id": null,
    "revision_number": 2,
    "security_groups": [
      "65db6fc3-7ba5-42e9-95f1-d45145f1e65f"
    ],
    "status": "DOWN",
    "qos_network_policy_id": null,
  }
}
```

```
"tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",  
"updated_at": "2021-02-24T07:01:15Z"  
}  
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 获取虚拟网卡

### 功能介绍

获取一个虚拟网卡的详情

### URI

```
GET /v2.0/ports/{port_id}
```

| 参数      | 是否必选 | 描述      |
|---------|------|---------|
| port_id | 是    | 虚拟网卡的ID |

### 请求消息

无

### 响应消息

| 参数             | 参数类型    | 描述                            |
|----------------|---------|-------------------------------|
| ports          | array   | 网卡列表。                         |
| admin_state_up | boolean | 网卡的 admin 状态，取值 true 或 false。 |



| 参数                      | 参数类型   | 描述                                                                                                   |
|-------------------------|--------|------------------------------------------------------------------------------------------------------|
| allowed_addresses_pairs | array  | 网卡允许访问的地址 pair 列表，每个 pair 包含一个 单独IP 或一个 IP 和 MAC。                                                    |
| binding:host_id         | string | 网卡所属的 host ID。                                                                                       |
| binding:profile         | object | 网卡信息字典，传递和接收用于后端网 络插件处理的网卡信息。                                                                        |
| binding:vif_details     | object | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。                                                      |
| binding:vif_type        | string | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap, 未绑定是 unbound，失败是 binding_failed。                           |
| binding:vnictype        | string | 网卡挂载的类型，比如 normal, direct, baremetal。                                                                |
| created_at              | string | 网卡创建时间 (in UTC ISO8601 format)。                                                                      |
| description             | string | 用户描述。                                                                                                |
| device_id               | string | 使用该网卡的设备 ID，比如一个虚拟机 uuid。                                                                            |
| device_owner            | string | 使用该网卡的设备类型，比如 network:distributed表示 metadata port, network:vip 表示虚拟 IP, network:secondary 表示辅助私网 IP。 |
| extra_dhcp_options      | array  | 额外的 DHCP 参数列表。                                                                                       |
| fixed_ips               | array  | 网卡的固定 IP 列表。                                                                                         |
| id                      | string | 网卡的 ID。                                                                                              |
| mac_address             | string | 网卡的 MAC 地址。                                                                                          |
| name                    | string | 网卡的名称。                                                                                               |
| network_id              | string | 网卡所属的网络 ID。                                                                                          |

| 参数                    | 参数类型    | 描述                                                                 |
|-----------------------|---------|--------------------------------------------------------------------|
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。 |
| project_id            | string  | 网卡所属的 project ID。                                                  |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                   |
| revision_number       | integer | 网卡的版本号。                                                            |
| security_groups       | array   | 网卡关联的安全组列表。                                                        |
| status                | string  | 网卡的状态：ACTIVE, DOWN, BUILD and ERROR。                               |
| tags                  | array   | 网卡的标签列表。                                                           |
| qos_network_policy_id | string  | 网卡所属network绑定的 QoS policy ID。                                      |
| tenant_id             | string  | 网卡所属的 tenant ID。                                                   |
| updated_at            | string  | 网卡最近更新的时间(in UTC ISO8601 format)。                                  |

## 请求示例

```
GET https://{endpoint}/v2.0/ports/01a9c14a-d4bb-4b28-a5c8-fd0458b8c94e
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": false,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnictype": "normal",
```

```
"created_at": "2021-02-22T12:29:22Z",
"description": "",
"device_id": "",
"device_owner": "",
"extra_dhcp_opts": [],
"fixed_ips": [
  {
    "ip_address": "172.16.0.9",
    "subnet_id": "c12ac01a-2422-473a-b297-baca99b2d690"
  }
],
"id": "01a9c14a-d4bb-4b28-a5c8-fd0458b8c94e",
"mac_address": "fa:16:3e:01:74:c7",
"name": "kw-2_kw-test_1edaecca",
"network_id": "328878f3-5805-42f1-82dd-40dcdd1f79f1",
"port_security_enabled": true,
"project_id": "8cd7d3e275584638964ff92ddd3b1596",
"qos_policy_id": null,
"revision_number": 5,
"security_groups": [
  "dbd9b12d-1d79-4594-ba18-36967bc4b9ae"
],
"status": "DOWN",
"qos_network_policy_id": null,
"tenant_id": "8cd7d3e275584638964ff92ddd3b1596",
"updated_at": "2021-02-22T12:29:22Z"
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 更新虚拟网卡

## 功能介绍

更新一个指定虚拟网卡

## URI

```
PUT /v2.0/ports/{port_id}
```

| 参数      | 是否必选 | 描述      |
|---------|------|---------|
| port_id | 是    | 虚拟网卡的ID |

## 请求消息

| 参数                      | 参数类型    | 是否必选 | 描述                                                                                                  |
|-------------------------|---------|------|-----------------------------------------------------------------------------------------------------|
| port                    | object  | 是    | 网卡对象。                                                                                               |
| admin_state_up          | boolean | 否    | 网卡的admin状态，取值true或false。                                                                            |
| allowed_addresses_pairs | array   | 否    | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个IP 和 MAC 。                                                   |
| binding:host_id         | string  | 否    | 网卡所属的 host ID。                                                                                      |
| binding:profile         | object  | 否    | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                                                        |
| binding:vnic_type       | string  | 否    | 网卡挂载的类型，比如normal，direct，baremetal，默认normal。                                                         |
| description             | string  | 否    | 用户描述。                                                                                               |
| device_id               | string  | 否    | 使用该网卡的设备ID，比如一个虚拟机uuid。                                                                             |
| device_owner            | string  | 否    | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port，network:vip 表示虚拟 IP，network:secondary 表示辅助私网 IP。 |

| 参数                    | 参数类型    | 是否必选 | 描述                                                           |
|-----------------------|---------|------|--------------------------------------------------------------|
| extra_dhcp_opts       | array   | 否    | 额外的 DHCP 参数列表。                                               |
| fixed_ips             | array   | 否    | 网卡的固定 IP 列表。                                                 |
| mac_address           | string  | 否    | 网卡的 MAC 地址。                                                  |
| name                  | string  | 否    | 网卡的名称。                                                       |
| port_security_enabled | boolean | 否    | 网卡的安全状态。true或者false，如果是true将会配置安全组规则和anti-spoofing规则，否则不会处理。 |
| qos_policy_id         | string  | 否    | 网卡关联的 QoS 策略 ID。                                             |
| security_groups       | array   | 否    | 网卡关联的安全组列表。                                                  |

## 响应消息

| 参数                      | 参数类型    | 描述                                                                 |
|-------------------------|---------|--------------------------------------------------------------------|
| port                    | object  | 网卡对象。                                                              |
| admin_state_up          | boolean | 网卡的admin状态，取值true或false。                                           |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC 。                 |
| binding:host_id         | string  | 网卡所属的 host ID。                                                     |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                                       |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是port_filter和ovs_hybrid_plug。                       |
| binding:vif_type        | string  | 网卡绑定使用的ml2插件类型，比如ovs，bridge，macvtap，未绑定是unbound，失败是binding_failed。 |

| 参数                    | 参数类型    | 描述                                                                                                  |
|-----------------------|---------|-----------------------------------------------------------------------------------------------------|
| binding:vnic_type     | string  | 网卡挂载的类型，比如normal，direct，baremetal。                                                                  |
| created_at            | string  | 网卡创建时间 (UTC ISO8601格式)。                                                                             |
| description           | string  | 用户描述。                                                                                               |
| device_id             | sttring | 使用该网卡的设备ID，比如一个虚机uuid。                                                                              |
| device_owner          | sting   | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port，network:vip 表示虚拟 IP，network:secondary 表示辅助私网 IP。 |
| extra_dhcp_opts       | array   | 额外的 DHCP 参数列表。                                                                                      |
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                                                        |
| id                    | string  | 网卡的ID。                                                                                              |
| mac_address           | string  | 网卡的 MAC 地址。                                                                                         |
| name                  | satring | 网卡的名称。                                                                                              |
| network_id            | string  | 网卡所属的网络ID。                                                                                          |
| port_security_enabled | boolean | 网卡的安全状态。true或者false，如果是true将会配置安全组规则和anti-spoofing规则，否则不会处理。                                        |
| project_id            | string  | 网卡所属的 project ID。                                                                                   |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                                                    |
| revision_number       | integer | 网卡的版本号。                                                                                             |
| security_groups       | array   | 网卡关联的安全组列表。                                                                                         |
| status                | string  | 网卡的状态： ACTIVE, DOWN, BUILD and ERROR。                                                               |
| tags                  | array   | 网卡的标签列表。                                                                                            |

| 参数                    | 参数类型   | 描述                            |
|-----------------------|--------|-------------------------------|
| qos_network_policy_id | string | 网卡所属network绑定的 QoS policy ID。 |
| tenant_id             | string | 网卡所属的 tenant ID。              |
| updated_at            | string | 网卡最近更新的时间(UTC ISO8601 格式)。    |

## 请求示例

```
PUT http://{endpoint}/v2.0/ports/602debb5-289d-4ecc-bc86-6a2f84a4b677
```

```
{
  "port": {
    "fixed_ips": [
      {
        "ip_address": "172.16.0.115",
        "subnet_id": "95347c6c-181d-4739-998d-0365487c9ac6"
      }
    ]
  }
}
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnic_type": "normal",
    "created_at": "2021-02-24T07:01:14Z",
    "description": "",
    "device_id": ""
  }
}
```

```
{
  "device_owner": "",
  "extra_dhcp_opts": [],
  "fixed_ips": [
    {
      "ip_address": "172.16.0.115",
      "subnet_id": "95347c6c-181d-4739-998d-0365487c9ac6"
    }
  ],
  "id": "602debb5-289d-4ecc-bc86-6a2f84a4b677",
  "mac_address": "fa:16:3e:91:6b:15",
  "name": "test",
  "network_id": "1191a2fd-42c6-44b9-b153-3378407468f2",
  "port_security_enabled": true,
  "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
  "qos_policy_id": null,
  "revision_number": 8,
  "security_groups": [
    "65db6fc3-7ba5-42e9-95f1-d45145f1e65f"
  ],
  "status": "DOWN",
  "qos_network_policy_id": null,
  "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
  "updated_at": "2021-02-24T08:35:26Z"
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 增加虚拟网卡的allowed\_address\_pairs

### 功能介绍

给指定虚拟网卡增加allowed\_address\_pairs

### URI



```
PUT /v2.0/ports/{port_id}/add_allowed_address_pairs
```

| 参数      | 是否必选 | 描述      |
|---------|------|---------|
| port_id | 是    | 虚拟网卡的ID |

## 请求消息

| 参数                      | 参数类型   | 是否必选 | 描述                                                                              |
|-------------------------|--------|------|---------------------------------------------------------------------------------|
| port                    | object | 是    | 网卡对象。                                                                           |
| allowed_addresses_pairs | array  | 否    | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC，辅助私网IP和虚拟IP每个 pair 使用单独一个IP即可。 |

## 响应消息

| 参数                      | 参数类型    | 描述                                                |
|-------------------------|---------|---------------------------------------------------|
| port                    | object  | 网卡对象。                                             |
| admin_state_up          | boolean | 网卡的 admin 状态，取值 true 或 false。                     |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC。 |
| binding:host_id         | string  | 网卡所属的 host ID。                                    |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                      |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。   |

| 参数                    | 参数类型    | 描述                                                                                                    |
|-----------------------|---------|-------------------------------------------------------------------------------------------------------|
| binding:vif_type      | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap, 未绑定是 unbound, 失败是 binding_failed。                           |
| binding:vnic_type     | string  | 网卡挂载的类型，比如normal, direct, baremetal。                                                                  |
| created_at            | string  | 网卡创建时间 (UTC ISO8601格式)。                                                                               |
| description           | string  | 用户描述。                                                                                                 |
| device_id             | string  | 使用该网卡的设备ID，比如一个虚拟机uuid。                                                                               |
| device_owner          | string  | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port, network:vip 表示虚拟 IP, network:secondary 表示辅助私网 IP。 |
| extra_dhcp_opts       | array   | 额外的 DHCP 参数列表。                                                                                        |
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                                                          |
| id                    | string  | 网卡的ID。                                                                                                |
| mac_address           | string  | 网卡的 MAC 地址。                                                                                           |
| name                  | string  | 网卡的名称。                                                                                                |
| network_id            | string  | 网卡所属的网络ID。                                                                                            |
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。                                    |
| project_id            | string  | 网卡所属的 project ID。                                                                                     |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                                                      |
| revision_number       | integer | 网卡的版本号。                                                                                               |
| security_groups       | array   | 网卡关联的安全组列表。                                                                                           |
| status                | string  | 网卡的状态： ACTIVE, DOWN, BUILD and ERROR。                                                                 |

| 参数                    | 参数类型   | 描述                              |
|-----------------------|--------|---------------------------------|
| tags                  | array  | 网卡的标签列表。                        |
| qos_network_policy_id | string | 网卡所属 network 绑定的 QoS policy ID。 |
| tenant_id             | string | 网卡所属的 tenant ID。                |
| updated_at            | string | 网卡最近更新的时间(UTC ISO8601 格式)。      |

## 请求示例

```
PUT http://{endpoint}/v2.0/ports/602debb5-289d-4ecc-bc86-6a2f84a4b677/add_allowed_address_pairs
```

```
{
  "port": {
    "allowed_address_pairs": [
      {
        "ip_address": "192.168.13.7",
        "mac_address": "fa:16:3e:09:1f:b2"
      }
    ]
  }
}
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnic_type": "normal",
  }
}
```

```
"created_at": "2021-02-24T07:01:14Z",
"description": "",
"device_id": "",
"device_owner": "",
"extra_dhcp_opts": [],
"fixed_ips": [
  {
    "ip_address": "192.168.13.7",
    "mac_address": "fa:16:3e:09:1f:b2"
  }
],
"id": "602debb5-289d-4ecc-bc86-6a2f84a4b677",
"mac_address": "fa:16:3e:91:6b:15",
"name": "test",
"network_id": "1191a2fd-42c6-44b9-b153-3378407468f2",
"port_security_enabled": true,
"project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
"qos_policy_id": null,
"revision_number": 8,
"security_groups": [
  "65db6fc3-7ba5-42e9-95f1-d45145f1e65f"
],
"status": "DOWN",
"qos_network_policy_id": null,
"tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
"updated_at": "2021-02-24T08:35:26Z"
}
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 移除虚拟网卡的allowed\_address\_pairs

## 功能介绍

给指定虚拟网卡移除allowed\_address\_pairs

## URI

```
PUT /v2.0/ports/{port_id}/remove_allowed_address_pairs
```

| 参数      | 是否必选 | 描述      |
|---------|------|---------|
| port_id | 是    | 虚拟网卡的ID |

## 请求消息

| 参数                      | 参数类型   | 是否必选 | 描述                                                |
|-------------------------|--------|------|---------------------------------------------------|
| port                    | object | 是    | 网卡对象。                                             |
| allowed_addresses_pairs | array  | 否    | 网卡允许访问的地址 pair 列表，每个pair 包含一个单独 IP 或一个 IP 和 MAC 。 |

## 响应消息

| 参数                      | 参数类型    | 描述                                                 |
|-------------------------|---------|----------------------------------------------------|
| port                    | object  | 网卡对象。                                              |
| admin_state_up          | boolean | 网卡的 admin 状态，取值 true 或 false。                      |
| allowed_addresses_pairs | array   | 网卡允许访问的地址 pair 列表，每个 pair 包含一个单独 IP 或一个 IP 和 MAC 。 |
| binding:host_id         | string  | 网卡所属的 host ID。                                     |
| binding:profile         | object  | 网卡信息字典，传递和接收用于后端网络插件处理的网卡信息。                       |
| binding:vif_details     | object  | 网卡附加信息字典，目前主要字段是 port_filter 和 ovs_hybrid_plug。    |

| 参数                    | 参数类型    | 描述                                                                                                    |
|-----------------------|---------|-------------------------------------------------------------------------------------------------------|
| binding:vif_type      | string  | 网卡绑定使用的 ml2 插件类型，比如 ovs, bridge, macvtap, 未绑定是 unbound, 失败是 binding_failed。                           |
| binding:vnic_type     | string  | 网卡挂载的类型，比如 normal, direct, baremetal。                                                                 |
| created_at            | string  | 网卡创建时间 (UTC ISO8601格式)。                                                                               |
| description           | string  | 用户描述。                                                                                                 |
| device_id             | string  | 使用该网卡的设备 ID，比如一个虚机 uuid。                                                                              |
| device_owner          | string  | 使用该网卡的设备类型，比如 network:distributed 表示 metadata port, network:vip 表示虚拟 IP, network:secondary 表示辅助私网 IP。 |
| extra_dhcp_opts       | array   | 额外的 DHCP 参数列表。                                                                                        |
| fixed_ips             | array   | 网卡的固定 IP 列表。                                                                                          |
| id                    | string  | 网卡的 ID。                                                                                               |
| mac_address           | string  | 网卡的 MAC 地址。                                                                                           |
| name                  | string  | 网卡的名称。                                                                                                |
| network_id            | string  | 网卡所属的网络 ID。                                                                                           |
| port_security_enabled | boolean | 网卡的安全状态。true 或者 false，如果是 true 将会配置安全组规则和 anti-spoofing 规则，否则不会处理。                                    |
| project_id            | string  | 网卡所属的 project ID。                                                                                     |
| qos_policy_id         | string  | 网卡关联的 QoS 策略 ID。                                                                                      |
| revision_number       | integer | 网卡的版本号。                                                                                               |
| security_groups       | array   | 网卡关联的安全组列表。                                                                                           |
| status                | string  | 网卡的状态：ACTIVE, DOWN, BUILD and ERROR。                                                                  |

| 参数                    | 参数类型   | 描述                              |
|-----------------------|--------|---------------------------------|
| tags                  | array  | 网卡的标签列表。                        |
| qos_network_policy_id | string | 网卡所属 network 绑定的 QoS policy ID。 |
| tenant_id             | string | 网卡所属的 tenant ID。                |
| updated_at            | string | 网卡最近更新的时间(UTC ISO8601 格式)。      |

## 请求示例

```
PUT http://{endpoint}/v2.0/ports/602debb5-289d-4ecc-bc86-6a2f84a4b677/remove_allowed_address_pairs
```

```
{
  "port": {
    "allowed_address_pairs": [
      {
        "ip_address": "192.168.13.7",
        "mac_address": "fa:16:3e:09:1f:b2"
      }
    ]
  }
}
```

## 正常响应示例

```
{
  "port": {
    "admin_state_up": true,
    "allowed_address_pairs": [],
    "binding:host_id": "",
    "binding:profile": {},
    "binding:vif_details": {},
    "binding:vif_type": "unbound",
    "binding:vnic_type": "normal",
  }
}
```

```
"created_at": "2021-02-24T07:01:14Z",
"description": "",
"device_id": "",
"device_owner": "",
"extra_dhcp_opts": [],
"fixed_ips": [
  {
    "ip_address": "192.168.13.7",
    "mac_address": "fa:16:3e:09:1f:b2"
  }
],
"id": "602debb5-289d-4ecc-bc86-6a2f84a4b677",
"mac_address": "fa:16:3e:91:6b:15",
"name": "test",
"network_id": "1191a2fd-42c6-44b9-b153-3378407468f2",
"port_security_enabled": true,
"project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
"qos_policy_id": null,
"revision_number": 8,
"security_groups": [
  "65db6fc3-7ba5-42e9-95f1-d45145f1e65f"
],
"status": "DOWN",
"qos_network_policy_id": null,
"tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
"updated_at": "2021-02-24T08:35:26Z"
}
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 删除虚拟网卡

## 功能介绍



删除一个指定虚拟网卡

## URI

```
DELETE /v2.0/ports/{port_id}
```

| 参数      | 是否必选 | 描述      |
|---------|------|---------|
| port_id | 是    | 虚拟网卡的ID |

## 请求消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/ports/01a9c14a-d4bb-4b28-a5c8-fd0458b8c94e
```

## 正常响应代码

204

## 错误码

400, 401, 403, 404, 409, 413

## 1.6 QoS策略

### QoS策略

#### 列举QoS policy

##### 功能介绍

列举QoS策略。

##### URI

```
GET /v2.0/qos/policies
```

##### 请求消息

无

##### 响应消息

| 参数              | 参数类型    | 描述          |
|-----------------|---------|-------------|
| policies        | array   | policy对象列表。 |
| description     | string  | 描述。         |
| tenant_id       | string  | 租户id。       |
| project_id      | string  | 租户id。       |
| revision_number | integer | 版本号。        |
| shared          | boolean | 是否租户项目间共享。  |
| id              | string  | QoS策略id。    |
| is_default      | boolean | 是否是默认策略。    |
| rules           | array   | QoS策略规则列表。  |

| 参数         | 参数类型   | 描述       |
|------------|--------|----------|
| name       | string | 名字。      |
| created_at | string | 创建时间。    |
| updated_at | string | 上一次更新时间。 |

## 请求示例

```
GET http://{endpoint}/v2.0/qos/policies
```

## 正常响应示例

```
{
  "policies": [
    {
      "name": "096c7e01-28ab-46e5-9427-1444f6f67a12",
      "rules": [
        {
          "max_kbps": 1024,
          "type": "bandwidth_limit",
          "id": "feaba4ba-c002-455c-810c-ffdaddac9c6a",
          "max_burst_kbps": 819,
          "qos_policy_id": "c15ce4a2-787e-4225-82f7-5ba538f0715e"
        }
      ],
      "tenant_id": "2caf92e24f284a76be9cb54577ecebed",
      "created_at": "2021-02-25T06:04:43Z",
      "updated_at": "2021-02-25T06:04:43Z",
      "revision_number": 2,
      "shared": false,
      "project_id": "2caf92e24f284a76be9cb54577ecebed",
      "id": "c15ce4a2-787e-4225-82f7-5ba538f0715e",
      "description": ""
    },
    {
      "name": "096c7e01-28ab-46e5-9427-1444f6f67a12",
      "rules": [
```

```
{
  "max_kbps":2048,
  "type":"bandwidth_limit",
  "id":"04cf4639-ffe8-42f3-bf79-42d22ccccfcf2",
  "max_burst_kbps":1638,
  "qos_policy_id":"ffc9b0cd-f4db-4db3-ac82-9be3401e3a53"
},
{
  "tenant_id":"2caf92e24f284a76be9cb54577ecebed",
  "created_at":"2021-02-25T06:27:57Z",
  "updated_at":"2021-02-25T06:27:57Z",
  "revision_number":2,
  "shared":false,
  "project_id":"2caf92e24f284a76be9cb54577ecebed",
  "id":"ffc9b0cd-f4db-4db3-ac82-9be3401e3a53",
  "description":""
}
]
```

## 正常响应代码

200

## 错误码

401

## 创建QoS policy

### 功能介绍

创建QoS策略。

### URI

```
POST /v2.0/qos/policies
```

### 请求消息

| 参数            | 参数类型 | 是否必选   | Description |
|---------------|------|--------|-------------|
| policy        | 是    | object | QoS策略对象。    |
| policy:name   | 是    | string | QoS策略名字。    |
| policy:shared | 否    | bool   | 是否共享。       |

## 响应消息

| 参数              | 参数类型    | 描述         |
|-----------------|---------|------------|
| policy          | object  | policy对象。  |
| description     | string  | 描述。        |
| tenant_id       | string  | 租户id。      |
| project_id      | string  | 租户id。      |
| revision_number | integer | 版本号。       |
| shared          | boolean | 是否租户项目间共享。 |
| id              | string  | QoS策略id。   |
| rules           | array   | QoS策略规则列表。 |
| name            | string  | 名字。        |
| created_at      | string  | 创建时间。      |
| updated_at      | string  | 上一次更新时间。   |
| is_default      | boolean | 是否是默认策略。   |

## 请求示例

```
POST http://{endpoint}/v2.0/qos/policies
```

```
{
  "policy":{
    "name":"test_qos"
  }
}
```

## 正常响应示例

```
{
  "policy":{
    "name":"test_qos",
    "rules":[

    ],
    "tenant_id":"cf8d21b7fc084813a0481fb9bf331357",
    "created_at":"2021-02-25T07:06:23Z",
    "updated_at":"2021-02-25T07:06:23Z",
    "revision_number":1,
    "shared":false,
    "project_id":"cf8d21b7fc084813a0481fb9bf331357",
    "id":"05e0ab13-0e6e-40d2-a04d-b5ead44d2653",
    "is_default": false,
    "description":""
  }
}
```

## 正常响应代码

201

## 错误码

401, 403, 404, 409

## 显示QoS policy详情

## 功能介绍

显示QoS策略详情。

## URI

```
GET /v2.0/qos/policies/{policy_id}
```

| 参数        | 参数类型   | 是否必选 | Description |
|-----------|--------|------|-------------|
| policy_id | string | 是    | QoS策略的ID。   |

## 请求消息

无

## 响应消息

| 参数              | 参数类型    | 描述         |
|-----------------|---------|------------|
| policy          | object  | policy对象。  |
| description     | string  | 描述。        |
| tenant_id       | string  | 租户id。      |
| project_id      | string  | 租户id。      |
| revision_number | integer | 版本。        |
| shared          | boolean | 是否租户项目间共享。 |
| id              | string  | QoS策略id。   |
| rules           | array   | QoS策略规则列表。 |
| name            | string  | 名字。        |
| created_at      | string  | 创建时间。      |
| updated_at      | string  | 上一次更新时间。   |
| is_default      | boolean | 是否是默认策略。   |

## 请求示例

```
GET http://{endpoint}/v2.0/qos/policies/ffc9b0cd-f4db-4db3-ac82-9be3401e3a53
```

## 正常响应示例

```
{
  "policy": {
    "name": "096c7e01-28ab-46e5-9427-1444f6f67a12",
    "rules": [
      {
        "max_kbps": 2048,
        "type": "bandwidth_limit",
        "id": "04cf4639-ffe8-42f3-bf79-42d22cccfcf2",
        "max_burst_kbps": 1638,
        "qos_policy_id": "ffc9b0cd-f4db-4db3-ac82-9be3401e3a53"
      }
    ],
    "tenant_id": "2caf92e24f284a76be9cb54577ecebed",
    "created_at": "2021-02-25T06:27:57Z",
    "updated_at": "2021-02-25T06:27:57Z",
    "revision_number": 2,
    "shared": false,
    "project_id": "2caf92e24f284a76be9cb54577ecebed",
    "id": "ffc9b0cd-f4db-4db3-ac82-9be3401e3a53",
    "is_default": false,
    "description": ""
  }
}
```

## 正常响应代码

200

## 错误码

401, 404

## 给QoS policy设置带宽



## 功能介绍

给QoS策略设置带宽规则。

## URI

```
POST /v2.0/qos/policies/{policy_id}/bandwidth_limit_rules
```

| 参数        | 参数类型   | 是否必选 | Description   |
|-----------|--------|------|---------------|
| policy_id | string | 否    | QoS policy id |

## 请求消息

| 参数                                  | 参数类型 | 是否必选    | Description                            |
|-------------------------------------|------|---------|----------------------------------------|
| bandwidth_limit_rule                | 是    | object  | 带宽限制对象。                                |
| bandwidth_limit_rule:max_kbps       | 否    | integer | 每秒最大流量(K)。                             |
| bandwidth_limit_rule:max_burst_kbps | 否    | integer | 流量峰值(K)。                               |
| bandwidth_limit_rule:direction      | 否    | string  | QoS规则方向 egress or ingress, 默认是 egress。 |

## 响应消息

| 参数                   | 参数类型    | 描述          |
|----------------------|---------|-------------|
| bandwidth_limit_rule | object  | 带宽限制规则对象。   |
| max_kbps             | integer | 每秒最大数据量(K)。 |
| id                   | string  | 带宽规则id。     |

| 参数             | 参数类型    | 描述                        |
|----------------|---------|---------------------------|
| max_burst_kbps | integer | 最大峰值流量(K)。                |
| direction      | string  | 方向ingress(入口),egress(出口)。 |

## 请求示例

```
POST http://{endpoint}/v2.0/qos/policies/05e0ab13-0e6e-40d2-a04d-b5ead44d2653/bandwidth_limit_rules
```

```
{
  "bandwidth_limit_rule": {
    "max_kbps": "2048",
    "max_burst_kbps": "2048"
  }
}
```

## 正常响应示例

```
{
  "bandwidth_limit_rule": {
    "id": "5f126d84-551a-4dcf-bb01-0e9c0df0c793",
    "max_kbps": 2048,
    "max_burst_kbps": 2048,
    "direction": "egress"
  }
}
```

## 正常响应代码

201

## 错误码

400, 401, 404, 409

## 更新QoS policy带宽

## 功能介绍

更新QoS带宽规则。

## URI

```
PUT /v2.0/qos/policies/{policy_id}/bandwidth_limit_rules/{rule_id}
```

| 参数        | 参数类型   | 是否必选 | Description |
|-----------|--------|------|-------------|
| policy_id | string | 是    | QoS策略的 id   |
| rule_id   | string | 是    | QoS规则的 id   |

## 请求消息

| 参数                                  | 参数类型 | 是否必选    | Description                            |
|-------------------------------------|------|---------|----------------------------------------|
| bandwidth_limit_rule                | 是    | object  | 带宽限制对象。                                |
| bandwidth_limit_rule:max_kbps       | 否    | integer | 每秒最大流量(K)。                             |
| bandwidth_limit_rule:max_burst_kbps | 否    | integer | 流量峰值(K)。                               |
| bandwidth_limit_rule:direction      | 否    | string  | QoS规则方向 egress or ingress, 默认是 egress。 |

## 响应消息

| 参数                   | 参数类型    | 描述          |
|----------------------|---------|-------------|
| bandwidth_limit_rule | object  | 带宽限制规则对象。   |
| max_kbps             | integer | 每秒最大数据量(K)。 |

| 参数             | 参数类型    | 描述                        |
|----------------|---------|---------------------------|
| id             | string  | 带宽规则id。                   |
| max_burst_kbps | integer | 最大峰值流量(K)。                |
| direction      | string  | 方向ingress(入口),egress(出口)。 |

## 请求示例

```
PUT http://{endpoint}/v2.0/qos/policies/05e0ab13-0e6e-40d2-a04d-b5ead44d2653/bandwidth_limit_rules/d1a8dc46-34df-453c-908c-a4eb36e0ba4f
```

```
{
  "bandwidth_limit_rule": {
    "max_kbps": "2048",
    "max_burst_kbps": "3096"
  }
}
```

## 正常响应示例

```
{
  "bandwidth_limit_rule": {
    "max_kbps": 2048,
    "id": "d1a8dc46-34df-453c-908c-a4eb36e0ba4f",
    "max_burst_kbps": 3096,
    "direction": "ingress"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 404

## 删除QoS policy带宽规则

### 功能介绍

删除QoS带宽规则。

### URI

```
DELETE /v2.0/qos/policies/{policy_id}/bandwidth_limit_rules/{rule_id}
```

| 参数        | 参数类型   | 是否必选 | Description |
|-----------|--------|------|-------------|
| policy_id | string | 是    | QoS策略的 id。  |
| rule_id   | string | 是    | QoS规则的 id。  |

### 请求消息

无

### 响应消息

无

### 请求示例

```
DELETE https://{endpoint}/v2.0/qos/policies/05e0ab13-0e6e-40d2-a04d-b5ead44d2653/bandwidth_limit_rules/d1a8dc46-34df-453c-908c-a4eb36e0ba4f
```

### 正常响应示例

无

### 正常响应代码

204

### 错误码

400, 401, 404

## 1.7 路由器

### 路由器

#### 列举路由器

##### 功能介绍

列举路由器

##### URI

```
GET /v2.0/routers
```

##### 请求消息

| 参数         | 参数类型   | 是否必选 | 描述          |
|------------|--------|------|-------------|
| project_id | string | 否    | 指定项目ID进行查询。 |

##### 响应消息

| 参数             | 参数类型    | 描述                            |
|----------------|---------|-------------------------------|
| routers        | array   | 路由器对象列表。                      |
| id             | string  | 路由器 id。                       |
| tenant_id      | string  | 租户id。                         |
| project_id     | string  | 租户id。                         |
| name           | string  | 名字。                           |
| description    | string  | 描述。                           |
| admin_state_up | boolean | admin状态, up/down(true/false)。 |

| 参数                      | 参数类型    | 描述                                                         |
|-------------------------|---------|------------------------------------------------------------|
| status                  | string  | 路由器状态。                                                     |
| external_gateway_info   | object  | 路由网关信息，需要有netwrok_id, enable_snat和external_fixed_ip否则字段为空。 |
| revision_number         | integer | 路由器的版本号。                                                   |
| routes                  | array   | 静态路由配置，内包含destination(目的地址)和nexthop(下一跳地址)。                |
| availability_zone_hints | array   | 路由可用域范围。                                                   |
| availability_zones      | array   | 路由所在的可用域。                                                  |
| created_at              | string  | 创建时间。                                                      |
| updated_at              | string  | 上一次更新时间。                                                   |

## 请求示例

```
GET https://{endpoint}/v2.0/routers
```

## 正常响应示例

```
{
  "routers": [
    {
      "status": "ACTIVE",
      "external_gateway_info": null,
      "availability_zone_hints": [
        "default-az"
      ],
      "availability_zones": [
        "default-az"
      ],
      "description": ""
    }
  ]
}
```

```
{
  "admin_state_up": true,
  "tenant_id": "870c5414baee43a4ba7adfb253e46018",
  "created_at": "2021-02-24T07:53:09Z",
  "updated_at": "2021-02-24T07:53:20Z",
  "project_id": "870c5414baee43a4ba7adfb253e46018",
  "revision_number": 8,
  "routes": [

  ],
  "id": "95405096-26e5-4f68-a952-5533c5f08292",
  "name": "test_router"
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 创建路由器

### 功能介绍

创建一个路由器

### URI

POST /v2.0/routers

### 请求消息

| 参数     | 参数类型   | 是否必选 | 描述     |
|--------|--------|------|--------|
| router | object | 是    | 路由器对象。 |



| 参数                      | 参数类型    | 是否必选 | 描述                        |
|-------------------------|---------|------|---------------------------|
| admin_state_up          | boolean | 否    | 路由器的admin状态，取值true或false。 |
| availability_zone_hints | array   | 否    | 路由可用域范围。                  |
| description             | string  | 否    | 用户描述。                     |
| name                    | string  | 否    | 路由器的名称。                   |
| project_id              | string  | 否    | 路由器所属的project id。         |
| tenant_id               | string  | 否    | 路由器所属的tenant id。          |

## 响应消息

| 参数                    | 参数类型    | 描述                                                        |
|-----------------------|---------|-----------------------------------------------------------|
| router                | object  | 路由器对象。                                                    |
| id                    | string  | 路由器 id。                                                   |
| tenant_id             | string  | 租户id。                                                     |
| project_id            | string  | 租户id。                                                     |
| name                  | string  | 名字。                                                       |
| description           | string  | 描述。                                                       |
| admin_state_up        | boolean | admin状态，up/down(true/false)。                              |
| status                | string  | 路由器状态。                                                    |
| external_gateway_info | object  | 路由网关信息，需要有network_id，enable_snat和external_fixed_ip否则字段为空。 |
| revision_number       | integer | 路由器的版本号。                                                  |

| 参数                      | 参数类型   | 描述                                          |
|-------------------------|--------|---------------------------------------------|
| routes                  | array  | 静态路由配置，内包含destination(目的地址)和nexthop(下一跳地址)。 |
| availability_zone_hints | array  | 路由可用域范围。                                    |
| availability_zones      | array  | 路由所在的可用域。                                   |
| created_at              | string | 创建时间。                                       |
| updated_at              | string | 上一次更新时间。                                    |

## 请求示例

```
POST http://{endpoint}/v2.0/routers
```

```
{
  "router": {
    "admin_state_up": true,
    "name": "test_router",
    "availability_zone_hints": [
      "default-az"
    ]
  }
}
```

## 正常响应示例

```
{
  "router": {
    "status": "ACTIVE",
    "external_gateway_info": null,
    "availability_zone_hints": [
      "default-az"
    ]
  }
}
```

```
    ],
    "availability_zones": [
      "default-az"
    ],
    "description": "",
    "admin_state_up": true,
    "tenant_id": "870c5414baee43a4ba7adfb253e46018",
    "created_at": "2021-02-24T07:53:09Z",
    "updated_at": "2021-02-24T07:53:20Z",
    "project_id": "870c5414baee43a4ba7adfb253e46018",
    "revision_number": 8,
    "routes": [

    ],
    "id": "95405096-26e5-4f68-a952-5533c5f08292",
    "name": "test_router"
  }
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 获取路由器

### 功能介绍

获取一个路由器的详情

### URI

```
GET /v2.0/routers/{router_id}
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

## 请求消息

无

## 响应消息

| 参数                      | 参数类型    | 描述                                                          |
|-------------------------|---------|-------------------------------------------------------------|
| router                  | object  | 路由器对象。                                                      |
| id                      | string  | 路由器 id。                                                     |
| tenant_id               | string  | 租户id。                                                       |
| project_id              | string  | 租户id。                                                       |
| name                    | string  | 名字。                                                         |
| description             | string  | 描述。                                                         |
| admin_state_up          | boolean | admin状态, up/down(true/false)。                               |
| status                  | string  | 路由器状态。                                                      |
| external_gateway_info   | object  | 路由网关信息, 需要有netwrok_id, enable_snat和external_fixed_ip否则字段为空。 |
| revision_number         | integer | 路由器的版本号。                                                    |
| routes                  | array   | 静态路由配置, 内包含destination(目的地址)和nexthop(下一跳地址)。                |
| availability_zone_hints | array   | 路由可用域范围。                                                    |
| availability_zones      | array   | 路由所在的可用域。                                                   |
| created_at              | string  | 创建时间。                                                       |
| updated_at              | string  | 上一次更新时间。                                                    |

## 请求示例

```
GET http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292
```

## 正常响应示例

```
{
  "router": {
    "status": "ACTIVE",
    "external_gateway_info": null,
    "availability_zone_hints": [
      "default-az"
    ],
    "availability_zones": [
      "default-az"
    ],
    "description": "",
    "admin_state_up": true,
    "tenant_id": "870c5414baee43a4ba7adfb253e46018",
    "created_at": "2021-02-24T07:53:09Z",
    "updated_at": "2021-02-24T07:53:20Z",
    "project_id": "870c5414baee43a4ba7adfb253e46018",
    "revision_number": 8,
    "routes": [

    ],
    "id": "95405096-26e5-4f68-a952-5533c5f08292",
    "name": "test_router"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 更新路由器

### 功能介绍

更新一个指定路由器

### URI

```
PUT /v2.0/routers/{router_id}
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

### 请求消息

| 参数             | 参数类型    | 是否必选 | 描述                                          |
|----------------|---------|------|---------------------------------------------|
| router         | object  | 是    | 路由器对象。                                      |
| admin_state_up | boolean | 否    | 路由器的admin状态，取值true或false。                   |
| description    | string  | 否    | 用户描述。                                       |
| name           | string  | 否    | 路由器的名称。                                     |
| routes         | array   | 否    | 静态路由配置，内包含destination(目的地址)和nexthop(下一跳地址)。 |

### 响应消息

| 参数     | 参数类型   | 描述         |
|--------|--------|------------|
| router | object | router对象。  |
| id     | string | router id。 |

| 参数                      | 参数类型    | 描述                                                          |
|-------------------------|---------|-------------------------------------------------------------|
| tenant_id               | string  | 租户id。                                                       |
| project_id              | string  | 租户id。                                                       |
| name                    | string  | 名字。                                                         |
| description             | string  | 描述。                                                         |
| admin_state_up          | boolean | admin状态, up/down(true/false)。                               |
| status                  | string  | 路由器状态。                                                      |
| external_gateway_info   | object  | 路由网关信息, 需要有netwrok_id, enable_snat和external_fixed_ip否则字段为空。 |
| revision_number         | integer | 路由器的版本号。                                                    |
| routes                  | array   | 静态路由配置, 内包含destination(目的地址)和nexthop(下一跳地址)。                |
| availability_zone_hints | array   | 路由可用域范围。                                                    |
| availability_zones      | array   | 路由所在的可用域。                                                   |
| created_at              | string  | 创建时间。                                                       |
| updated_at              | string  | 上一次更新时间。                                                    |

## 请求示例

设置静态路由

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292
```

```
{
  "router": {
    "routes": {
```

```

        "destination": "192.168.112.0/24",
        "nexthop": "192.168.111.7"
    }
}

```

## 正常响应示例

```

{
  "router": {
    "status": "ACTIVE",
    "external_gateway_info": null,
    "availability_zone_hints": [
      "default-az"
    ],
    "availability_zones": [
      "default-az"
    ],
    "description": "",
    "admin_state_up": true,
    "tenant_id": "870c5414baee43a4ba7adfb253e46018",
    "created_at": "2021-02-24T07:53:09Z",
    "updated_at": "2021-02-24T07:53:20Z",
    "project_id": "870c5414baee43a4ba7adfb253e46018",
    "revision_number": 8,
    "routes": [
      {
        "destination": "192.168.112.0/24",
        "nexthop": "192.168.111.7"
      }
    ],
    "id": "95405096-26e5-4f68-a952-5533c5f08292",
    "name": "test_router"
  }
}

```

## 正常响应代码

200



## 错误码

400, 401, 403, 404, 409, 413

## 删除路由器

### 功能介绍

删除一个指定路由器

### URI

```
DELETE /v2.0/routers/{router_id}
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

### 请求消息

无

### 请求示例

```
DELETE http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292
```

### 正常响应代码

204

### 错误码

400, 401, 403, 404, 409, 413

## 路由器设置网关

### 功能介绍

给一个路由器设置外部网关

## URI

```
PUT /v2.0/routers/{router_id}
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

## 请求消息

| 参数                    | 参数类型   | 是否必选 | 描述                                                                                    |
|-----------------------|--------|------|---------------------------------------------------------------------------------------|
| router                | object | 是    | 路由器对象。                                                                                |
| external_gateway_info | object | 是    | 外部网络信息，必须指定network_id，可选项external_fixed_ips网关ip，rate_limit带宽，enable_snat是否开启snat默认开启。 |

## 响应消息

| 参数             | 参数类型    | 描述                           |
|----------------|---------|------------------------------|
| router         | object  | 路由器对象。                       |
| id             | string  | 路由器 id。                      |
| tenant_id      | string  | 租户id。                        |
| project_id     | string  | 租户id。                        |
| name           | string  | 名字。                          |
| description    | string  | 描述。                          |
| admin_state_up | boolean | admin状态，up/down(true/false)。 |

| 参数                      | 参数类型    | 描述                                                         |
|-------------------------|---------|------------------------------------------------------------|
| status                  | string  | 路由器状态。                                                     |
| external_gateway_info   | object  | 路由网关信息，需要有network_id, enable_snat和external_fixed_ip否则字段为空。 |
| revision_number         | integer | 路由器的版本号。                                                   |
| routes                  | array   | 静态路由配置，内包含destination(目的地址)和nexthop(下一跳地址)。                |
| availability_zone_hints | array   | 路由可用域范围。                                                   |
| availability_zones      | array   | 路由所在的可用域。                                                  |
| created_at              | string  | 创建时间。                                                      |
| updated_at              | string  | 上一次更新时间。                                                   |

## 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292
```

```
{
  "router":{
    "external_gateway_info":{
      "network_id":"2dd8b394-fa34-44e8-9ea7-0e470ee5bcd9",
      "external_fixed_ips": [
        {
          "ip_address": "172.16.10.130",
          "subnet_id": "08fe297b-db20-4b46-9341-a6812c9f748f"
        }
      ],
      "rate_limit":1024000
    }
  }
}
```

## 正常响应示例

```
{
  "router": {
    "status": "ACTIVE",
    "external_gateway_info": {
      "enable_snat": true,
      "external_fixed_ips": [
        {
          "ip_address": "172.16.10.130",
          "subnet_id": "08fe297b-db20-4b46-9341-a6812c9f748f"
        }
      ],
      "network_id": "2dd8b394-fa34-44e8-9ea7-0e470ee5bcd9",
      "rate_limit": 1024000
    },
    "availability_zone_hints": [
      "default-az"
    ],
    "availability_zones": [
      "default-az"
    ],
    "description": "",
    "admin_state_up": true,
    "tenant_id": "870c5414baee43a4ba7adfb253e46018",
    "created_at": "2021-02-24T07:53:09Z",
    "updated_at": "2021-02-24T07:53:20Z",
    "project_id": "870c5414baee43a4ba7adfb253e46018",
    "revision_number": 8,
    "routes": [
      {
        "destination": "192.168.112.0/24",
        "nexthop": "192.168.111.7"
      }
    ],
    "id": "95405096-26e5-4f68-a952-5533c5f08292",
    "name": "test_router"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 路由器清除网关

### 功能介绍

清除一个路由器的外部网关

### URI

```
PUT /v2.0/routers/{router_id}
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

### 请求消息

| 参数                    | 参数类型   | 是否必选 | 描述                |
|-----------------------|--------|------|-------------------|
| router                | object | 是    | 路由器对象。            |
| external_gateway_info | object | 是    | 外部网络信息，此API中传空字典。 |

### 响应消息

| 参数 | 参数类型 | 描述 |
|----|------|----|
|----|------|----|

| 参数                      | 参数类型    | 描述                                                          |
|-------------------------|---------|-------------------------------------------------------------|
| router                  | object  | 路由器对象。                                                      |
| id                      | string  | 路由器 id。                                                     |
| tenant_id               | string  | 租户id。                                                       |
| project_id              | string  | 租户id。                                                       |
| name                    | string  | 名字。                                                         |
| description             | string  | 描述。                                                         |
| admin_state_up          | boolean | admin状态, up/down(true/false)。                               |
| status                  | string  | 路由器状态。                                                      |
| external_gateway_info   | object  | 路由网关信息, 需要有netwrok_id, enable_snat和external_fixed_ip否则字段为空。 |
| revision_number         | integer | 路由器的版本号。                                                    |
| routes                  | array   | 静态路由配置, 内包含destination(目的地址)和nexthop(下一跳地址)。                |
| availability_zone_hints | array   | 路由可用域范围。                                                    |
| availability_zones      | array   | 路由所在的可用域。                                                   |
| created_at              | string  | 创建时间。                                                       |
| updated_at              | string  | 上一次更新时间。                                                    |

## 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292
```

```
{
  "router":{
    "external_gateway_info":{

    }
  }
}
```

## 正常响应示例

```
{
  "router":
  {
    "status":"ACTIVE",
    "external_gateway_info": null,
    "availability_zone_hints":[
      "default-az"
    ],
    "availability_zones":[
      "default-az"
    ],
    "description":"",
    "admin_state_up":true,
    "tenant_id":"870c5414baee43a4ba7adfb253e46018",
    "created_at":"2021-02-24T07:53:09Z",
    "updated_at":"2021-02-24T07:53:20Z",
    "project_id":"870c5414baee43a4ba7adfb253e46018",
    "revision_number":8,
    "routes":[
      {
        "destination": "192.168.112.0/24",
        "nexthop": "192.168.111.7"
      }
    ],
    "id":"95405096-26e5-4f68-a952-5533c5f08292",
    "name":"test_router"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 路由器连接子网

### 功能介绍

给一个路由器连接子网

### URI

```
PUT /v2.0/routers/{router_id}/add_router_interface
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

### 请求消息

| 参数        | 参数类型   | 是否必选                       | 描述            |
|-----------|--------|----------------------------|---------------|
| subnet_id | string | 是(subnet_id或者port_id两个二选一) | 连接的子网的uuid。   |
| port_id   | string | 是                          | 子网中port的uuid。 |

### 响应消息

| 参数     | 参数类型   | 描述      |
|--------|--------|---------|
| router | object | 路由器对象。  |
| id     | string | 路由器 id。 |



| 参数         | 参数类型   | 描述              |
|------------|--------|-----------------|
| tenant_id  | string | 租户id。           |
| project_id | string | 租户id。           |
| network_id | string | 连接的子网所属网络的uuid。 |
| subnet_id  | string | 连接的子网的uuid。     |
| subnet_ids | array  | 连接的子网uuid列表。    |
| port_id    | string | 子网中port的uuid。   |

## 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292/add_router_interface
```

```
{
  "subnet_id": "8bb1d1e8-a479-4f4f-8ca1-3bc80eabfe47"
}
```

## 正常响应示例

```
{
  "id": "95405096-26e5-4f68-a952-5533c5f08292",
  "network_id": "a1bd29e8-9349-44a5-9fdf-39462c08d761",
  "port_id": "2b32e183-17fc-4d2f-bea9-355803bd9bc3",
  "subnet_id": "8bb1d1e8-a479-4f4f-8ca1-3bc80eabfe47",
  "subnet_ids": [
    "8bb1d1e8-a479-4f4f-8ca1-3bc80eabfe47"
  ],
  "tenant_id": "870c5414baee43a4ba7adfb253e46018"
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 路由器断开子网

### 功能介绍

断开一个路由器的某个子网

### URI

```
PUT /v2.0/routers/{router_id}/remove_router_interface
```

| 参数        | 是否必选 | 描述      |
|-----------|------|---------|
| router_id | 是    | 路由器的ID。 |

### 请求消息

| 参数        | 参数类型   | 是否必选                       | 描述            |
|-----------|--------|----------------------------|---------------|
| subnet_id | string | 是(subnet_id或者port_id两个二选一) | 连接的子网的uuid。   |
| port_id   | string | 是                          | 子网中port的uuid。 |

### 响应消息

| 参数         | 参数类型   | 描述      |
|------------|--------|---------|
| router     | object | 路由器对象。  |
| id         | string | 路由器 id。 |
| tenant_id  | string | 租户id。   |
| project_id | string | 租户id。   |

| 参数         | 参数类型   | 描述              |
|------------|--------|-----------------|
| network_id | string | 连接的子网所属网络的uuid。 |
| subnet_id  | string | 连接的子网的uuid。     |
| subnet_ids | array  | 连接的子网uuid列表。    |
| port_id    | string | 子网中port的uuid。   |

### 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292/remove_router_interface
```

```
{
  "subnet_id": "8bb1d1e8-a479-4f4f-8ca1-3bc80eabfe47"
}
```

### 正常响应示例

```
{
  "id": "95405096-26e5-4f68-a952-5533c5f08292",
  "network_id": "a1bd29e8-9349-44a5-9fdf-39462c08d761",
  "port_id": "2b32e183-17fc-4d2f-bea9-355803bd9bc3",
  "subnet_id": "8bb1d1e8-a479-4f4f-8ca1-3bc80eabfe47",
  "subnet_ids": [
    "8bb1d1e8-a479-4f4f-8ca1-3bc80eabfe47"
  ],
  "tenant_id": "870c5414baee43a4ba7adfb253e46018"
}
```

### 正常响应代码

200

### 错误码

400, 401, 403, 404, 409, 413

## 列举路由器静态路由

### 功能介绍

列举路由器的静态路由

### URI

```
GET /v2.0/routers/{router_id}/show_extraroutes
```

### 请求消息

| 参数        | 参数类型   | 是否必选 | 描述      |
|-----------|--------|------|---------|
| router_id | string | 是    | 路由器的ID。 |

### 响应消息

| 参数        | 参数类型   | 描述        |
|-----------|--------|-----------|
| router_id | string | 路由器 id。   |
| routes    | array  | 路由器的静态路由。 |

### 请求示例

```
GET https://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292/show_extraroutes
```

### 正常响应示例

```
{
  "router_id" : "95405096-26e5-4f68-a952-5533c5f08292",
  "routes" : [
```

```
{ "destination" : "10.0.1.0/24", "nexthop" : "10.0.0.11",  
  "description" : "test" },  
  { "destination" : "10.0.2.0/24", "nexthop" : "10.0.0.12",  
    "description" : "test" }  
]
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 新增路由器静态路由

## 功能介绍

新增路由器的静态路由

## URI

```
PUT /v2.0/routers/{router_id}/add_extraroutes
```

## 请求消息

| 参数        | 参数类型   | 是否必选 | 描述                                                               |
|-----------|--------|------|------------------------------------------------------------------|
| router_id | string | 是    | 路由器的ID。                                                          |
| routes    | array  | 是    | 需要新增的静态路由。每条静态路由必须包含两个参数：destination和nexthop，可选一个参数：description。 |

## 响应消息

| 参数 | 参数类型 | 描述 |
|----|------|----|
|----|------|----|

| 参数     | 参数类型   | 描述        |
|--------|--------|-----------|
| id     | string | 路由器 ID。   |
| name   | string | 路由器的 名字。  |
| routes | array  | 路由器的静态路由。 |

## 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292/add_extraroutes
```

```
{
  "router" : {
    "routes" : [
      { "destination" : "10.0.3.0/24", "nexthop" : "10.0.0.13",
"description" : "test" },
      { "destination" : "10.0.4.0/24", "nexthop" : "10.0.0.14",
"description" : "test" }
    ]
  }
}
```

## 正常响应示例

```
{
  "router" : {
    "id" : "95405096-26e5-4f68-a952-5533c5f08292",
    "name" : "router1",
    "routes" : [
      { "destination" : "10.0.3.0/24", "nexthop" : "10.0.0.13",
"description" : "test" },
      { "destination" : "10.0.4.0/24", "nexthop" : "10.0.0.14",
"description" : "test" }
    ]
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 更新路由器静态路由

### 功能介绍

更新路由器一条静态路由的description

### URI

```
PUT /v2.0/routers/{router_id}/update_extraroute_attr
```

### 请求消息

| 参数        | 是否必选 | 描述                                                      |
|-----------|------|---------------------------------------------------------|
| router_id | 是    | 路由器的ID。                                                 |
| route     | 是    | 需要更新的静态路由。静态路由必须包含三个参数：destination、nexthop、description。 |

### 响应消息

| 参数        | 参数类型   | 描述        |
|-----------|--------|-----------|
| router_id | string | 路由器 ID。   |
| route     | object | 更新后的静态路由。 |

### 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292/update_extraroute_attr
```

```
{
  "destination": "192.168.112.0/24",
  "nexthop": "192.168.111.7",
  "description" : "test"
}
```

## 正常响应示例

```
{
  "router_id" : "95405096-26e5-4f68-a952-5533c5f08292",
  "route" : {
    "destination": "192.168.112.0/24",
    "nexthop": "192.168.111.7",
    "description" : "test"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 删除路由器静态路由

### 功能介绍

删除路由器的静态路由

### URI

```
PUT /v2.0/routers/{router_id}/remove_extraroutes
```



| 参数        | 是否必选 | 描述                                          |
|-----------|------|---------------------------------------------|
| router_id | 是    | 路由器的ID。                                     |
| routes    | 是    | 需要删除的静态路由。每条静态路由包含两个参数：destination和nexthop。 |

## 响应消息

| 参数     | 参数类型   | 描述           |
|--------|--------|--------------|
| id     | string | 路由器 id。      |
| name   | string | 路由器的名字。      |
| routes | array  | 删除后路由器的静态路由。 |

## 请求示例

```
PUT http://{endpoint}/v2.0/routers/95405096-26e5-4f68-a952-5533c5f08292/remove_extraroutes
```

```
{
  "router" : {
    "routes" : [
      { "destination" : "10.0.3.0/24", "nexthop" : "10.0.0.13" },
      { "destination" : "10.0.4.0/24", "nexthop" : "10.0.0.14" }
    ]
  }
}
```

## 正常响应示例

```
{
  "router" : {
    "id" : "95405096-26e5-4f68-a952-5533c5f08292",
  }
}
```

```
    "name" : "router1",  
    "routes" : []  
  }  
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 1.8 策略路由

### 策略路由

#### 列举路由器

#### 功能介绍

列举策略路由

#### URI

```
GET /v2.0/policyroutes
```

#### 请求消息

| 参数         | 参数类型   | 是否必选 | 描述           |
|------------|--------|------|--------------|
| project_id | string | 否    | 指定项目ID进行查询。  |
| router_id  | string | 否    | 指定路由器ID进行查询。 |

#### 响应消息

| 参数           | 参数类型   | 描述                                                                       |
|--------------|--------|--------------------------------------------------------------------------|
| policyroutes | array  | 策略路由列表。                                                                  |
| action       | string | 策略动作。取值包括：allow 允许，deny 拒绝，reroute 重新路由。其中 reroute 需要配合 nexthops 字段实现功能。 |
| created_at   | string | 创建时间。                                                                    |
| description  | string | 描述。                                                                      |
| destination  | string | 目标地址。                                                                    |

| 参数              | 参数类型    | 描述                                             |
|-----------------|---------|------------------------------------------------|
| dst_port        | string  | 目标端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。         |
| enabled         | boolean | 策略路由是否开启，取值： true 开启， false 禁用。                |
| ethertype       | string  | 以太网协议， IPv4 或 IPv6。                            |
| from_interface  | string  | 源接口的 UUID，必需属于绑定的路由器的接口。                       |
| id              | string  | 策略路由 ID。                                       |
| nexthops        | string  | reroute 重路由的下一跳 IP 地址。多个下一跳地址表示 ecmp 路由，用逗号隔开。 |
| priority        | integer | 策略路由的优先级。                                      |
| project_id      | string  | 租户 ID。                                         |
| protocol        | String  | IP 协议。取值包括: icmp、tcp、udp 或 null。               |
| revision_number | integer | 策略路由的版本号。                                      |
| router_id       | string  | 绑定的路由器 UUID。                                   |
| source          | string  | 源地址。                                           |
| src_port        | string  | 源端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。          |
| tags            | array   | 标签列表。                                          |
| tenant_id       | string  | 租户 ID。                                         |
| to_interface    | string  | 目标接口的 UUID，必需属于绑定的路由器的接口。                      |
| updated_at      | string  | 上一次更新时间。                                       |

## 请求示例

```
GET https://{endpoint}/v2.0/policyroutes?router_id=d536a8ba-6b98-4059-85a2-1c2c60014f60
```

## 正常响应示例

```
{
  "policyroutes": [
    {
      "action": "reroute",
      "created_at": "2024-12-24T06:07:55Z",
      "description": "",
      "destination": null,
      "dst_port": null,
      "enabled": true,
      "ethertype": "IPv4",
      "from_interface": "b2b3f0d0-cf6b-41a2-bd5e-5cfcf3b87f87",
      "id": "fb0c6e80-34aa-484b-b9c8-4a6908bd25ae",
      "nexthops": "172.16.10.23,172.16.10.24",
      "priority": 123,
      "project_id": "ed6255c8005846889398598becb9878f",
      "protocol": null,
      "revision_number": 0,
      "router_id": "d536a8ba-6b98-4059-85a2-1c2c60014f60",
      "source": "192.168.117.0/24",
      "src_port": null,
      "tags": [],
      "tenant_id": "ed6255c8005846889398598becb9878f",
      "to_interface": null,
      "updated_at": "2024-12-24T06:07:55Z"
    }
  ]
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 创建策略路由

### 功能介绍

创建一个策略路由

### URI

POST /v2.0/policyroutes

### 请求消息

| 参数             | 参数类型    | 是否必选 | 描述                                                                                   |
|----------------|---------|------|--------------------------------------------------------------------------------------|
| policyroute    | object  | 是    | 策略路由对象。                                                                              |
| action         | string  | 否    | 策略动作。取值包括：allow 允许，deny 拒绝，reroute 重新路由，默认是 reroute。其中 reroute 需要配合 nexthops 字段实现功能。 |
| description    | string  | 否    | 描述。                                                                                  |
| destination    | string  | 否    | 目标地址。                                                                                |
| dst_port       | string  | 否    | 目标端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。                                               |
| enabled        | boolean | 否    | 策略路由是否开启，取值：true 开启，false 禁用。默认是 true。                                               |
| ethertype      | string  | 否    | 以太网协议，IPv4 或 IPv6。默认是IPv4。                                                           |
| from_interface | string  | 否    | 源接口的 UUID，必需属于绑定的路由器的接口。                                                             |
| nexthops       | string  | 否    | reroute 重路由的下一跳 IP 地址。多个下一跳地址表示 ecmp 路由，用逗号隔开。                                       |

| 参数           | 参数类型    | 是否必选 | 描述                                    |
|--------------|---------|------|---------------------------------------|
| priority     | integer | 否    | 策略路由的优先级。                             |
| project_id   | string  | 否    | 租户 ID。                                |
| protocol     | string  | 否    | IP 协议。取值包括: icmp、tcp、udp 或 null。      |
| router_id    | string  | 是    | 绑定的路由器 UUID。                          |
| source       | string  | 否    | 源地址。                                  |
| src_port     | string  | 否    | 源端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。 |
| tenant_id    | string  | 否    | 租户 ID。                                |
| to_interface | string  | 否    | 目标接口的 UUID，必需属于绑定的路由器的接口。             |

## 响应消息

| 参数          | 参数类型    | 描述                                                                       |
|-------------|---------|--------------------------------------------------------------------------|
| action      | string  | 策略动作。取值包括：allow 允许，deny 拒绝，reroute 重新路由。其中 reroute 需要配合 nexthops 字段实现功能。 |
| created_at  | string  | 创建时间。                                                                    |
| description | string  | 描述。                                                                      |
| destination | string  | 目标地址。                                                                    |
| dst_port    | string  | 目标端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。                                   |
| enabled     | boolean | 策略路由是否开启，取值：true 开启，false 禁用。                                            |
| ethertype   | string  | 以太网协议，IPv4 或 IPv6。                                                       |

| 参数              | 参数类型    | 描述                                             |
|-----------------|---------|------------------------------------------------|
| from_interface  | string  | 源接口的 UUID，必需属于绑定的路由器的接口。                       |
| id              | string  | 策略路由 ID。                                       |
| nexthops        | string  | reroute 重路由的下一跳 IP 地址。多个下一跳地址表示 ecmp 路由，用逗号隔开。 |
| priority        | integer | 策略路由的优先级。                                      |
| project_id      | string  | 租户 ID。                                         |
| protocol        | String  | IP 协议。取值包括: icmp、tcp、udp 或 null。               |
| revision_number | integer | 策略路由的版本号。                                      |
| router_id       | string  | 绑定的路由器 UUID。                                   |
| source          | string  | 源地址。                                           |
| src_port        | string  | 源端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。          |
| tags            | array   | 标签列表。                                          |
| tenant_id       | string  | 租户 ID。                                         |
| to_interface    | string  | 目标接口的 UUID，必需属于绑定的路由器的接口。                      |
| updated_at      | string  | 上一次更新时间。                                       |

## 请求示例

```
POST http://{endpoint}/v2.0/policyroutes
```

```
{
  "policyroute": {
    "action": "reroute",
    "destination": "172.16.10.251/32",
```



```

    "dst_port": "22",
    "enabled": true,
    "ethertype": "IPv4",
    "from_interface": "b2b3f0d0-cf6b-41a2-bd5e-5cfcf3b87f87",
    "nexthops": "172.16.10.23,172.16.10.24",
    "priority": 123,
    "project_id": "ed6255c8005846889398598becb9878f",
    "protocol": tcp,
    "router_id": "d536a8ba-6b98-4059-85a2-1c2c60014f60",
    "source": "192.168.117.0/24",
    "src_port": "55:80",
    "to_interface": "e2369f70-077c-4a1e-a45e-bb88a7ee13fe",
  }
}

```

## 正常响应示例

```

{
  "policyroute": {
    "action": "reroute",
    "created_at": "2024-12-24T06:07:55Z",
    "description": "",
    "destination": "172.16.10.251/32",
    "dst_port": "22",
    "enabled": true,
    "ethertype": "IPv4",
    "from_interface": "b2b3f0d0-cf6b-41a2-bd5e-5cfcf3b87f87",
    "id": "fb0c6e80-34aa-484b-b9c8-4a6908bd25ae",
    "nexthops": "172.16.10.23,172.16.10.24",
    "priority": 123,
    "project_id": "ed6255c8005846889398598becb9878f",
    "protocol": tcp,
    "revision_number": 0,
    "router_id": "d536a8ba-6b98-4059-85a2-1c2c60014f60",
    "source": "192.168.117.0/24",
    "src_port": "55:80",
    "tags": [],
    "tenant_id": "ed6255c8005846889398598becb9878f",
    "to_interface": "e2369f70-077c-4a1e-a45e-bb88a7ee13fe",
    "updated_at": "2024-12-24T06:07:55Z"
  }
}

```

```
}  
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 获取策略路由

### 功能介绍

获取一个策略路由的详情

### URI

```
GET /v2.0/policyroutes/{policyroute_id}
```

| 参数             | 是否必选 | 描述        |
|----------------|------|-----------|
| policyroute_id | 是    | 策略路由的 ID。 |

### 请求消息

无

### 响应消息

| 参数          | 参数类型   | 描述                                                                       |
|-------------|--------|--------------------------------------------------------------------------|
| action      | string | 策略动作。取值包括：allow 允许，deny 拒绝，reroute 重新路由。其中 reroute 需要配合 nexthops 字段实现功能。 |
| created_at  | string | 创建时间。                                                                    |
| description | string | 描述。                                                                      |

| 参数              | 参数类型    | 描述                                             |
|-----------------|---------|------------------------------------------------|
| destination     | string  | 目标地址。                                          |
| dst_port        | string  | 目标端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。         |
| enabled         | boolean | 策略路由是否开启，取值： true 开启， false 禁用。                |
| ethertype       | string  | 以太网协议，IPv4 或 IPv6。                             |
| from_interface  | string  | 源接口的 UUID，必需属于绑定的路由器的接口。                       |
| id              | string  | 策略路由 ID。                                       |
| nexthops        | string  | reroute 重路由的下一跳 IP 地址。多个下一跳地址表示 ecmp 路由，用逗号隔开。 |
| priority        | integer | 策略路由的优先级。                                      |
| project_id      | string  | 租户 ID。                                         |
| protocol        | String  | IP 协议。取值包括: icmp、tcp、udp 或 null。               |
| revision_number | integer | 策略路由的版本号。                                      |
| router_id       | string  | 绑定的路由器 UUID。                                   |
| source          | string  | 源地址。                                           |
| src_port        | string  | 源端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。          |
| tags            | array   | 标签列表。                                          |
| tenant_id       | string  | 租户 ID。                                         |
| to_interface    | string  | 目标接口的 UUID，必需属于绑定的路由器的接口。                      |
| updated_at      | string  | 上一次更新时间。                                       |

## 请求示例

```
GET http://{endpoint}/v2.0/policyroutes/fb0c6e80-34aa-484b-b9c8-4a6908bd25ae
```

## 正常响应示例

```
{
  "policyroute": {
    "action": "reroute",
    "created_at": "2024-12-24T06:07:55Z",
    "description": "",
    "destination": "172.16.10.251/32",
    "dst_port": "22",
    "enabled": true,
    "ethertype": "IPv4",
    "from_interface": "b2b3f0d0-cf6b-41a2-bd5e-5cfcf3b87f87",
    "id": "fb0c6e80-34aa-484b-b9c8-4a6908bd25ae",
    "nexthops": "172.16.10.23,172.16.10.24",
    "priority": 123,
    "project_id": "ed6255c8005846889398598becb9878f",
    "protocol": tcp,
    "revision_number": 0,
    "router_id": "d536a8ba-6b98-4059-85a2-1c2c60014f60",
    "source": "192.168.117.0/24",
    "src_port": "55:80",
    "tags": [],
    "tenant_id": "ed6255c8005846889398598becb9878f",
    "to_interface": "e2369f70-077c-4a1e-a45e-bb88a7ee13fe",
    "updated_at": "2024-12-24T06:07:55Z"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 更新策略路由

## 功能介绍

更新一个指定策略路由

## URI

```
PUT /v2.0/policyroutes/{policyroute_id}
```

| 参数             | 是否必选 | 描述        |
|----------------|------|-----------|
| policyroute_id | 是    | 策略路由的 ID。 |

## 请求消息

| 参数             | 参数类型    | 是否必选 | 描述                                             |
|----------------|---------|------|------------------------------------------------|
| policyroute    | object  | 是    | 策略路由对象。                                        |
| description    | string  | 否    | 描述。                                            |
| destination    | string  | 否    | 目标地址。                                          |
| dst_port       | string  | 否    | 目标端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。         |
| enabled        | boolean | 否    | 策略路由是否开启，取值：true 开启，false 禁用。                  |
| from_interface | string  | 否    | 源接口的 UUID，必需属于绑定的路由器的接口。                       |
| nexthops       | string  | 否    | reroute 重路由的下一跳 IP 地址。多个下一跳地址表示 ecmp 路由，用逗号隔开。 |
| priority       | integer | 否    | 策略路由的优先级。                                      |
| protocol       | string  | 否    | IP 协议。取值包括：icmp、tcp、udp 或 null。                |
| router_id      | string  | 是    | 绑定的路由器 UUID。                                   |

| 参数           | 参数类型   | 是否必选 | 描述                                    |
|--------------|--------|------|---------------------------------------|
| source       | string | 否    | 源地址。                                  |
| src_port     | string | 否    | 源端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。 |
| to_interface | string | 否    | 目标接口的 UUID，必需属于绑定的路由器的接口。             |

## 响应消息

| 参数             | 参数类型    | 描述                                                                       |
|----------------|---------|--------------------------------------------------------------------------|
| action         | string  | 策略动作。取值包括：allow 允许，deny 拒绝，reroute 重新路由。其中 reroute 需要配合 nexthops 字段实现功能。 |
| created_at     | string  | 创建时间。                                                                    |
| description    | string  | 描述。                                                                      |
| destination    | string  | 目标地址。                                                                    |
| dst_port       | string  | 目标端口。有效的值可以是单个端口号，例如22；也可以是范围，例如80:90。                                   |
| enabled        | boolean | 策略路由是否开启，取值：true 开启，false 禁用。                                            |
| ethertype      | string  | 以太网协议，IPv4 或 IPv6。                                                       |
| from_interface | string  | 源接口的 UUID，必需属于绑定的路由器的接口。                                                 |
| id             | string  | 策略路由 ID。                                                                 |
| nexthops       | string  | reroute 重新路由的下一跳 IP 地址。多个下一跳地址表示 ECMP 路由，用逗号隔开。                          |
| priority       | integer | 策略路由的优先级。                                                                |
| project_id     | string  | 租户 ID。                                                                   |

| 参数              | 参数类型    | 描述                                       |
|-----------------|---------|------------------------------------------|
| protocol        | String  | IP 协议。取值包括: icmp、tcp、udp 或 null。         |
| revision_number | integer | 策略路由的版本号。                                |
| router_id       | string  | 绑定的路由器 UUID。                             |
| source          | string  | 源地址。                                     |
| src_port        | string  | 源端口。有效的值可以是单个端口号, 例如22; 也可以是范围, 例如80:90。 |
| tags            | array   | 标签列表。                                    |
| tenant_id       | string  | 租户 ID。                                   |
| to_interface    | string  | 目标接口的 UUID, 必需属于绑定的路由器的接口。               |
| updated_at      | string  | 上一次更新时间。                                 |

## 请求示例

```
PUT http://{endpoint}/v2.0/policyroutes/fb0c6e80-34aa-484b-b9c8-4a6908bd25ae
```

```
{
  "policyroute": {
    "destination": "172.16.10.251/32",
    "dst_port": "80",
    "from_interface": "b2b3f0d0-cf6b-41a2-bd5e-5cfcf3b87f87",
    "nexthops": "172.16.10.25,172.16.10.26",
    "project_id": "ed6255c8005846889398598becb9878f",
    "protocol": tcp,
    "router_id": "d536a8ba-6b98-4059-85a2-1c2c60014f60",
    "source": "192.168.117.0/24",
    "src_port": "55:800",
    "to_interface": "e2369f70-077c-4a1e-a45e-bb88a7ee13fe",
  }
}
```

## 正常响应示例

```
{
  "policyroute": {
    "action": "reroute",
    "created_at": "2024-12-24T06:07:55Z",
    "description": "",
    "destination": "172.16.10.251/32",
    "dst_port": "80",
    "enabled": true,
    "ethertype": "IPv4",
    "from_interface": "b2b3f0d0-cf6b-41a2-bd5e-5cfcf3b87f87",
    "id": "fb0c6e80-34aa-484b-b9c8-4a6908bd25ae",
    "nexthops": "172.16.10.25,172.16.10.26",
    "priority": 123,
    "project_id": "ed6255c8005846889398598becb9878f",
    "protocol": tcp,
    "revision_number": 0,
    "router_id": "d536a8ba-6b98-4059-85a2-1c2c60014f60",
    "source": "192.168.117.0/24",
    "src_port": "55:800",
    "tags": [],
    "tenant_id": "ed6255c8005846889398598becb9878f",
    "to_interface": "e2369f70-077c-4a1e-a45e-bb88a7ee13fe",
    "updated_at": "2024-12-24T08:07:55Z"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 删除策略路由

### 功能介绍

删除一个指定策略路由



## URI

```
DELETE /v2.0/policyroutes/{policyroute_id}
```

| 参数             | 是否必选 | 描述        |
|----------------|------|-----------|
| policyroute_id | 是    | 策略路由的 ID。 |

## 请求消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/policyroutes/fb0c6e80-34aa-484b-b9c8-4a6908bd25ae
```

## 正常响应代码

204

## 错误码

400, 401, 403, 404, 409, 413

## 1.9 安全组

### 安全组

#### 列举安全组

##### 功能介绍

列举网络安全组

##### URI

```
GET /v2.0/security-groups
```

##### 请求消息

| 参数         | 参数类型   | 是否必选 | 描述          |
|------------|--------|------|-------------|
| project_id | string | 否    | 指定项目ID进行查询。 |

##### 响应消息

| 参数              | 参数类型    | 描述                         |
|-----------------|---------|----------------------------|
| security-groups | array   | 列表组对象。                     |
| created_at      | string  | 安全组的创建时间 (UTC ISO8601 格式)。 |
| description     | string  | 用户描述。                      |
| id              | string  | 安全组的ID。                    |
| name            | string  | 安全组的名称。                    |
| project_id      | string  | 安全组所属的project id。          |
| revision_number | integer | 安全组的版本号。                   |

| 参数                   | 参数类型    | 描述                         |
|----------------------|---------|----------------------------|
| security_group_rules | array   | 安全组包含的规则列表。                |
| stateful             | boolean | 安全组是否为有状态。                 |
| tenant_id            | string  | 安全组所属的tenant id。           |
| updated_at           | string  | 安全组最近更新的时间(UTC ISO8601格式)。 |

## 请求示例

```
GET http://{endpoint}/v2.0/security-groups
```

## 正常响应示例

```
{
  "security_groups": [
    {
      "created_at": "2021-02-24T09:06:32Z",
      "description": "",
      "id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
      "name": "test",
      "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
      "revision_number": 1,
      "security_group_rules": [
        {
          "created_at": "2021-02-24T09:06:32Z",
          "description": null,
          "direction": "egress",
          "ethertype": "IPv4",
          "id": "231354cc-c516-47fd-9cc5-c20aab9b0e6c",
          "port_range_max": null,
          "port_range_min": null,
          "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
          "protocol": null,
          "remote_group_id": null,
          "remote_ip_prefix": null,
          "revision_number": 1,

```

```
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
    },
    {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv6",
        "id": "2455c213-22ec-4d8d-b668-483cf8f831b3",
        "port_range_max": null,
        "port_range_min": null,
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "revision_number": 1,
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
    }
],
"stateful": true,
"tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
"updated_at": "2021-02-24T09:06:32Z"
}
]
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 创建安全组

## 功能介绍

创建一个安全组

## URI

POST /v2.0/security-groups

## 请求消息

| 参数          | 参数类型           | 是否必选   | Description |
|-------------|----------------|--------|-------------|
|             | security-group | object | 是           |
| description | string         | 否      | 用户描述。       |
| name        | string         | 否      | 安全组的名称。     |
| stateful    | boolean        | 否      | 安全组是否为有状态。  |
| tenant_id   | string         | 否      | 所属租户ID。     |

## 响应消息

| 参数                   | 参数类型    | 描述                         |
|----------------------|---------|----------------------------|
| security-group       | object  | 安全组对象。                     |
| created_at           | string  | 安全组的创建时间 (UTC ISO8601 格式)。 |
| description          | string  | 用户描述。                      |
| id                   | string  | 安全组的ID。                    |
| name                 | string  | 安全组的名称。                    |
| project_id           | string  | 安全组所属的project id。          |
| revision_number      | integer | 安全组的版本号。                   |
| security_group_rules | array   | 安全组包含的规则列表。                |
| stateful             | boolean | 安全组是否为有状态。                 |

| 参数         | 参数类型   | 描述                         |
|------------|--------|----------------------------|
| tenant_id  | string | 安全组所属的tenant id。           |
| updated_at | string | 安全组最近更新的时间(UTC ISO8601格式)。 |

## 请求示例

```
POST http://{endpoint}/v2.0/security-groups
```

```
{
  "security_group": {
    "name": "test",
    "stateful": true,
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d"
  }
}
```

## 正常响应示例

```
{
  "security_group": {
    "created_at": "2021-02-24T09:06:32Z",
    "description": "",
    "id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "name": "test",
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "revision_number": 1,
    "security_group_rules": [
      {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "231354cc-c516-47fd-9cc5-c20aab9b0e6c",
        "port_range_max": null,
        "port_range_min": null,
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
```

```
        "protocol": null,  
        "remote_group_id": null,  
        "remote_ip_prefix": null,  
        "revision_number": 1,  
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",  
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",  
        "updated_at": "2021-02-24T09:06:32Z"  
    },  
    {  
        "created_at": "2021-02-24T09:06:32Z",  
        "description": null,  
        "direction": "egress",  
        "ethertype": "IPv6",  
        "id": "2455c213-22ec-4d8d-b668-483cf8f831b3",  
        "port_range_max": null,  
        "port_range_min": null,  
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",  
        "protocol": null,  
        "remote_group_id": null,  
        "remote_ip_prefix": null,  
        "revision_number": 1,  
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",  
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",  
        "updated_at": "2021-02-24T09:06:32Z"  
    }  
],  
    "stateful": true,  
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",  
    "updated_at": "2021-02-24T09:06:32Z"  
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 获取安全组

## 功能介绍

获取一个安全组的详情

## URI

```
GET /v2.0/security-groups/{security-group-id}
```

| 参数                | 是否必选 | 描述       |
|-------------------|------|----------|
| security-group-id | 是    | 安全组 的ID。 |

## 请求消息

无

## 响应消息

| 参数                   | 参数类型    | 描述                          |
|----------------------|---------|-----------------------------|
| security-group       | object  | 安全组对象。                      |
| created_at           | string  | 安全组的创建时间 (UTC ISO8601 格式)。  |
| description          | string  | 用户描述。                       |
| id                   | string  | 安全组的ID。                     |
| name                 | string  | 安全组的名称。                     |
| project_id           | string  | 安全组所属的project id。           |
| revision_number      | integer | 安全组的版本号。                    |
| security_group_rules | array   | 安全组包含的规则列表。                 |
| stateful             | boolean | 安全组是否为有状态。                  |
| tenant_id            | string  | 安全组所属的tenant id。            |
| updated_at           | string  | 安全组最近更新的时间(UTC ISO8601 格式)。 |



## 请求示例

```
GET http://{endpoint}/v2.0/ports/01a9c14a-d4bb-4b28-a5c8-fd0458b8c94e
```

## 正常响应示例

```
{
  "security_group": {
    "created_at": "2021-02-24T09:06:32Z",
    "description": "",
    "id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "name": "test",
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "revision_number": 1,
    "security_group_rules": [
      {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "231354cc-c516-47fd-9cc5-c20aab9b0e6c",
        "port_range_max": null,
        "port_range_min": null,
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "revision_number": 1,
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
      },
      {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv6",
        "id": "2455c213-22ec-4d8d-b668-483cf8f831b3",
        "port_range_max": null,
        "port_range_min": null,
```

```
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "revision_number": 1,
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
    },
    ],
    "stateful": true,
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "updated_at": "2021-02-24T09:06:32Z"
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 更新安全组

### 功能介绍

更新一个指定安全组

### URI

```
PUT /v2.0/security-groups/{security-group-id}
```

| 参数                | 是否必选 | 描述      |
|-------------------|------|---------|
| security-group-id | 是    | 安全组的ID。 |

### 请求消息

| 参数             | 参数类型    | 是否必选 | 描述         |
|----------------|---------|------|------------|
| security-group | object  | 是    | 安全组对象。     |
| description    | string  | 否    | 用户描述。      |
| stateful       | boolean | 否    | 安全组是否为有状态。 |
| name           | string  | 否    | 安全组的名称。    |

## 响应消息

| 参数                   | 参数类型    | 描述                         |
|----------------------|---------|----------------------------|
| security-group       | object  | 安全组对象。                     |
| created_at           | string  | 安全组的创建时间 (UTC ISO8601 格式)。 |
| description          | string  | 用户描述。                      |
| id                   | string  | 安全组的ID。                    |
| name                 | string  | 安全组的名称。                    |
| project_id           | string  | 安全组所属的project id。          |
| revision_number      | integer | 安全组的版本号。                   |
| security_group_rules | array   | 安全组包含的规则列表。                |
| stateful             | boolean | 安全组是否为有状态。                 |
| tenant_id            | string  | 安全组所属的tenant id。           |
| updated_at           | string  | 安全组最近更新的时间(UTC ISO8601格式)。 |

## 请求示例

```
PUT http://{endpoint}/v2.0/security-groups/de80111e-4ddb-4b58-a295-0b32e9a4f963
```

```
{
  "security_group": {
    "name": "test2"
  }
}
```

## 正常响应示例

```
{
  "security_group": {
    "created_at": "2021-02-24T09:06:32Z",
    "description": "",
    "id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "name": "test2",
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "revision_number": 1,
    "security_group_rules": [
      {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "231354cc-c516-47fd-9cc5-c20aab9b0e6c",
        "port_range_max": null,
        "port_range_min": null,
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "revision_number": 1,
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
      },
      {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv6",
```

```
    "id": "2455c213-22ec-4d8d-b668-483cf8f831b3",
    "port_range_max": null,
    "port_range_min": null,
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "protocol": null,
    "remote_group_id": null,
    "remote_ip_prefix": null,
    "revision_number": 1,
    "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "updated_at": "2021-02-24T09:06:32Z"
  },
  ],
  "stateful": true,
  "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
  "updated_at": "2021-02-24T10:06:32Z"
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 删除安全组

### 功能介绍

删除一个指定安全组

### URI

```
DELETE /v2.0/security-groups/{security-group-id}
```

| 参数                | 是否必选 | 描述      |
|-------------------|------|---------|
| security-group-id | 是    | 安全组的ID。 |

## 请求消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/security-groups/de80111e-4ddb-4b58-a295-0b32e9a4f963
```

## 正常响应代码

204

## 错误码

400, 401, 403, 404, 409, 413

## 列举安全组规则

### 功能介绍

列举所有安全组规则

### URI

```
GET /v2.0/security-group-rules
```

## 请求消息

无

## 响应消息

| 参数                   | 参数类型   | 描述                          |
|----------------------|--------|-----------------------------|
| security-group-rules | array  | 安全组规则列表。                    |
| created_at           | string | 安全组规则的创建时间 (UTC ISO8601格式)。 |
| description          | string | 用户描述。                       |

| 参数                | 参数类型    | 描述                                                 |
|-------------------|---------|----------------------------------------------------|
| direction         | string  | 安全组规则作用的方向，ingress或egress。                         |
| ethertype         | string  | 匹配的以太网协议，必需是IPv4或IPv6。                             |
| id                | string  | 安全组规则的ID。                                          |
| project_id        | string  | 安全组规则所。属的project id。                               |
| protocol          | string  | 匹配的IP协议类型，比如ICMP type。                             |
| port_range_max    | integer | 匹配的协议端口号最大值,如果协议是ICMP，则表示ICMP code。                |
| port_range_min    | integer | 匹配的协议端口号最小值,如果协议是ICMP，则表示ICMP code。                |
| remote_ip_prefix  | string  | 匹配的远端ip前缀。                                         |
| remote_group_id   | string  | 匹配的远端安全组的ID，可以指定remote_group_id或者remote_ip_prefix。 |
| revision_number   | integer | 安全组规则的版本号。                                         |
| security_group_id | string  | 关联的安全组ID。                                          |
| normalized_cidr   | string  | 只读的字段，规范化cidr的值。                                   |
| tenant_id         | string  | 安全组规则所属的tenant id。                                 |
| updated_at        | string  | 安全组规则最近更新的时间(UTC ISO8601格式)。                       |

## 请求示例

```
GET http://{endpoint}/v2.0/security-group-rules
```

## 正常响应示例

```
{
  "security_group_rules": [
    {
      "created_at": "2021-02-22T03:23:55Z",
      "description": null,
      "direction": "egress",
      "ethertype": "IPv4",
      "id": "00c1068e-0110-4cf2-8856-0e20268562fd",
      "port_range_max": null,
      "port_range_min": null,
      "project_id": "6e3cf7a9d62a4faf8162cacfc94126a9",
      "protocol": null,
      "remote_group_id": null,
      "remote_ip_prefix": null,
      "revision_number": 1,
      "normalized_cidr": null,
      "security_group_id": "61884e28-ca98-4e76-b0f3-6d5b3536bd7b",
      "tenant_id": "6e3cf7a9d62a4faf8162cacfc94126a9",
      "updated_at": "2021-02-22T03:23:55Z"
    },
    {
      "created_at": "2021-02-25T03:07:27Z",
      "description": "",
      "direction": "ingress",
      "ethertype": "IPv4",
      "id": "0184bfe7-f838-476f-ad19-6891215583f5",
      "port_range_max": 65535,
      "port_range_min": 1,
      "project_id": "e9f2f863c96841e49ba4e9946ebe9087",
      "protocol": "tcp",
      "remote_group_id": null,
      "remote_ip_prefix": "0.0.0.0/0",
      "revision_number": 1,
      "normalized_cidr": null,
      "security_group_id": "a12e08af-9e8d-461c-952d-3c6fbc1bebcbb",
      "tenant_id": "e9f2f863c96841e49ba4e9946ebe9087",
      "updated_at": "2021-02-25T03:07:27Z"
    }
  ]
}
```



## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 创建安全组规则

### 功能介绍

创建一个安全组规则

### URI

POST /v2.0/security-group-rules

### 请求消息

| 参数                  | 参数类型    | 是否必选 | Description                          |
|---------------------|---------|------|--------------------------------------|
| security-group-rule | object  | 是    | 安全组规则对象。                             |
| description         | string  | 否    | 用户描述。                                |
| direction           | string  | 是    | 安全组规则作用的方向, ingress或egress。          |
| ethertype           | string  | 否    | 匹配的以太网协议类型, 必需是IPv4或IPv6。            |
| port_range_max      | integer | 否    | 匹配的协议端口号最大值,如果协议是ICMP, 则表示ICMP code。 |
| port_range_min      | integer | 否    | 匹配的协议端口号最小值,如果协议是ICMP, 则表示ICMP code。 |
| protocol            | string  | 否    | 匹配的IP协议类型, 比如ICMP type。              |

| 参数                | 参数类型   | 是否必选 | Description                                        |
|-------------------|--------|------|----------------------------------------------------|
| remote_ip_prefix  | string | 否    | 匹配的远端ip前缀。                                         |
| remote_group_id   | string | 否    | 匹配的远端安全组ID, 可以指定remote_group_id或者remote_ip_prefix。 |
| security_group_id | string | 是    | 关联的安全组ID。                                          |
| tenant_id         | string | 否    | 所属租户ID。                                            |

## 响应消息

| 参数                  | 参数类型    | 描述                                   |
|---------------------|---------|--------------------------------------|
| security-group-rule | object  | 安全组规则。                               |
| created_at          | string  | 安全组规则的创建时间 (UTC ISO8601格式)。          |
| description         | string  | 用户描述。                                |
| direction           | string  | 安全组规则作用的方向, ingress或egress。          |
| ethertype           | string  | 匹配的以太网协议, 必需是IPv4或IPv6。              |
| id                  | string  | 安全组规则的ID。                            |
| project_id          | string  | 安全组规则所属的project id。                  |
| protocol            | string  | 匹配的IP协议类型, 比如ICMP type。              |
| port_range_max      | integer | 匹配的协议端口号最大值,如果协议是ICMP, 则表示ICMP code。 |
| port_range_min      | integer | 匹配的协议端口号最小值,如果协议是ICMP, 则表示ICMP code。 |

| 参数                | 参数类型    | 描述                                                 |
|-------------------|---------|----------------------------------------------------|
| remote_ip_prefix  | string  | 匹配的远端ip前缀。                                         |
| remote_group_id   | string  | 匹配的远端安全组的ID，可以指定remote_group_id或者remote_ip_prefix。 |
| revision_number   | integer | 安全组规则的版本号。                                         |
| security_group_id | string  | 关联的安全组ID。                                          |
| normalized_cidr   | string  | 只读的字段，规范化cidr的值。                                   |
| tenant_id         | string  | 安全组规则所属的tenant id。                                 |
| updated_at        | string  | 安全组规则最近更新的时间(UTC ISO8601格式)。                       |

### 请求示例

```
POST http://{endpoint}/v2.0/security-group-rules
```

```
{
  "security_group_rule": {
    "direction": "ingress",
    "ethertype": "IPv4",
    "port_range_max": "88",
    "port_range_min": "22",
    "protocol": "tcp",
    "remote_ip_prefix": "0.0.0.0/24",
    "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963"
  }
}
```

### 正常响应示例

```
{
  "security_group_rule": {
    "created_at": "2021-02-25T06:19:33Z",
    "description": "",
    "direction": "ingress",
    "ethertype": "IPv4",
    "id": "5dc48c92-e097-4aa4-ae9b-88550dd0170d",
    "port_range_max": 88,
    "port_range_min": 22,
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "protocol": "tcp",
    "remote_group_id": null,
    "remote_ip_prefix": "0.0.0.0/24",
    "revision_number": 1,
    "normalized_cidr": null,
    "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "updated_at": "2021-02-25T06:19:33Z"
  }
}
```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 获取安全组规则

### 功能介绍

获取一个安全组规则的详情

### URI

```
GET /v2.0/security-group-rules/{security-group-rule-id}
```

| 参数                     | 是否必选 | 描述        |
|------------------------|------|-----------|
| security-group-rule-id | 是    | 安全组规则的ID。 |

## 请求消息

无

## 响应消息

| 参数                  | 参数类型    | 描述                                                 |
|---------------------|---------|----------------------------------------------------|
| security-group-rule | object  | 安全组规则对象。                                           |
| created_at          | string  | 安全组规则的创建时间 (UTC ISO8601 form)。                     |
| description         | string  | 用户描述。                                              |
| direction           | string  | 安全组规则作用的方向，ingress或egress。                         |
| ethertype           | string  | 匹配的以太网协议，必需是IPv4或IPv6。                             |
| id                  | string  | 安全组规则的ID。                                          |
| project_id          | string  | 安全组规则所属的project id。                                |
| protocol            | string  | 匹配的IP协议类型，比如ICMP type。                             |
| port_range_max      | integer | 匹配的协议端口号最大值,如果协议是ICMP，则表示ICMP code。                |
| port_range_min      | integer | 匹配的协议端口号最小值,如果协议是ICMP，则表示ICMP code。                |
| remote_ip_prefix    | string  | 匹配的远端ip前缀。                                         |
| remote_group_id     | string  | 匹配的远端安全组的ID，可以指定remote_group_id或者remote_ip_prefix。 |

| 参数                | 参数类型    | 描述                           |
|-------------------|---------|------------------------------|
| revision_number   | integer | 安全组规则的版本号。                   |
| security_group_id | string  | 关联的安全组ID。                    |
| normalized_cidr   | string  | 只读的字段，规范化cidr的值。             |
| tenant_id         | string  | 安全组规则所属的tenant id。           |
| updated_at        | string  | 安全组规则最近更新的时间(UTC ISO8601格式)。 |

### 请求示例

```
GET http://{endpoint}/v2.0/security-group-rules/ 5dc48c92-e097-4aa4-ae9b-88550dd0170d
```

### 正常响应示例

```
{
  "security_group_rule": {
    "created_at": "2021-02-25T06:19:33Z",
    "description": "",
    "direction": "ingress",
    "ethertype": "IPv4",
    "id": "5dc48c92-e097-4aa4-ae9b-88550dd0170d",
    "port_range_max": 88,
    "port_range_min": 22,
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "protocol": "tcp",
    "remote_group_id": null,
    "remote_ip_prefix": "0.0.0.0/24",
    "revision_number": 1,
    "normalized_cidr": null,
    "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "updated_at": "2021-02-25T06:19:33Z"
  }
}
```

```
}  
}
```

## 正常响应代码

200

## 错误码

400, 401, 403, 404, 409, 413

## 删除安全组规则

### 功能介绍

删除一个指定安全组规则

### URI

```
DELETE /v2.0/security-group-rules/{security-group-rule-id}
```

| 参数                     | 是否必选 | 描述      |
|------------------------|------|---------|
| security-group-rule-id | 是    | 安全组的ID。 |

### 请求消息

无

### 请求示例

```
DELETE http://{endpoint}/v2.0/security-group-rules/ 5dc48c92-e097-4aa4-ae9b-88550dd0170d
```

## 正常响应代码

204

## 错误码

400, 401, 403, 404, 409, 413

## 克隆安全组

### 功能介绍

创建一个新的安全组，克隆指定安全组的所有规则

### URI

```
PUT /v2.0/security-groups/{security-group-id}/clone
```

### 请求消息

| 参数          | 参数类型           | 是否必选   | Description |
|-------------|----------------|--------|-------------|
|             | security-group | object | 是           |
| description | string         | 否      | 用户描述。       |
| name        | string         | 否      | 安全组的名称。     |
| stateful    | boolean        | 否      | 安全组是否为有状态。  |
| tenant_id   | string         | 否      | 所属租户ID。     |

### 响应消息

| 参数             | 参数类型   | 描述                         |
|----------------|--------|----------------------------|
| security-group | object | 安全组对象。                     |
| created_at     | string | 安全组的创建时间 (UTC ISO8601 格式)。 |
| description    | string | 用户描述。                      |
| id             | string | 安全组的ID。                    |
| name           | string | 安全组的名称。                    |



| 参数                   | 参数类型    | 描述                         |
|----------------------|---------|----------------------------|
| project_id           | string  | 安全组所属的project id。          |
| revision_number      | integer | 安全组的版本号。                   |
| security_group_rules | array   | 安全组包含的规则列表。                |
| stateful             | boolean | 安全组是否为有状态。                 |
| tenant_id            | string  | 安全组所属的tenant id。           |
| updated_at           | string  | 安全组最近更新的时间(UTC ISO8601格式)。 |

### 请求示例

```
PUT http://{endpoint}/v2.0/security-groups/5dc48c92-e097-4aa4-ae9b-88550dd0170d/clone
```

```
{
  "security_group": {
    "name": "clone_sg",
    "stateful": true,
    "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d"
  }
}
```

### 正常响应示例

```
{
  "security_group": {
    "created_at": "2021-02-24T09:06:32Z",
    "description": "",
    "id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
    "name": "clone_sg",
    "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
    "revision_number": 1,
    "security_group_rules": [
      {
```

```

        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv4",
        "id": "231354cc-c516-47fd-9cc5-c20aab9b0e6c",
        "port_range_max": null,
        "port_range_min": null,
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "revision_number": 1,
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
    },
    {
        "created_at": "2021-02-24T09:06:32Z",
        "description": null,
        "direction": "egress",
        "ethertype": "IPv6",
        "id": "2455c213-22ec-4d8d-b668-483cf8f831b3",
        "port_range_max": null,
        "port_range_min": null,
        "project_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "protocol": null,
        "remote_group_id": null,
        "remote_ip_prefix": null,
        "revision_number": 1,
        "security_group_id": "de80111e-4ddb-4b58-a295-0b32e9a4f963",
        "tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
        "updated_at": "2021-02-24T09:06:32Z"
    }
],
"stateful": true,
"tenant_id": "8dac1bf1c3a24be18809f82e1f9a845d",
"updated_at": "2021-02-24T09:06:32Z"
}
}

```

## 正常响应代码

201

## 错误码

400, 401, 403, 404, 409, 413

## 1.10 网络ACL/网关防火墙

网络ACL和网关防火墙共同使用防火墙API，即firewall\_groups，根据firewall\_groups关联的ports类型来区别是网络ACL还是网关防火墙，其中port类型包括子网网关port和路由器网关port，每个防火墙绑定的port类型必须一致。

- 当防火墙绑定的port类型为子网网关port时，该防火墙为网络ACL
- 当防火墙绑定的port类型为路由器网关port时，该防火墙为网关防火墙

此外，建议给网络ACL和网关防火墙打上标签。

- 为网络ACL打上"NetworkACL"标签，即URL为 `PUT /v2.0/firewall_groups/{firewall_group_id}/tags/NetworkACL`
- 为网关防火墙打上"GatewayFirewall"标签，即URL为 `PUT /v2.0/firewall_groups/{firewall_group_id}/tags/GatewayFirewall`

### 查询防火墙列表

#### 功能介绍

查询防火墙列表

#### URI

```
GET /v2.0/fwaas/firewall_groups
```

#### 请求消息

| 参数 | 参数类型 | 是否必选 | 描述 |
|----|------|------|----|
|----|------|------|----|

| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数                         | 参数类型    | 描述                                                             |
|----------------------------|---------|----------------------------------------------------------------|
| firewall_groups            | object  | firewall_group对象列表。                                            |
| admin_state_up             | boolean | 防火墙的管理状态，为up (true)或down (false)。默认是true。                      |
| description                | String  | 防火墙的描述。                                                        |
| id                         | String  | 防火墙ID。                                                         |
| egress_firewall_policy_id  | String  | 防火墙的egress策略ID。                                                |
| ingress_firewall_policy_id | String  | 防火墙的ingress策略ID。                                               |
| name                       | String  | 防火墙名称。                                                         |
| tags                       | array   | 资源的标记列表。                                                       |
| ports                      | array   | 与防火墙相关联的端口id列表。                                                |
| shared                     | boolean | 指示此防火墙是否在所有项目之间共享。                                             |
| status                     | String  | 防火墙状态。有效值为ACTIVE、INACTIVE、ERROR、PENDING_UPDATE或PENDING_DELETE。 |
| project_id                 | String  | 资源所属项目的ID。                                                     |

| 参数        | 参数类型   | 描述         |
|-----------|--------|------------|
| tenant_id | String | 资源所属项目的ID。 |

## 请求示例

```
GET http://{endpoint}/v2.0/fwaas/firewall_groups
```

## 正常响应示例

```
{
  "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": "",
      "tags": [],
      "ports": [
        "650bfd2f-7766-4a0d-839f-218f33e16998"
      ],
      "shared": true,
      "project_id": "45977fa2dbd7482098dd68d0d8970117",
      "status": "ACTIVE",
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

## 正常响应代码

200

## 错误码

401, 403

## 查询防火墙详情

### 功能介绍

查询防火墙详情

### URI

```
GET /v2.0/fwaas/firewall_groups/{firewall_group_id}
```

| 参数                | 是否必选 | 描述      |
|-------------------|------|---------|
| firewall_group_id | 是    | 防火墙的ID。 |

### 请求消息

无

### 响应消息

| 参数                         | 参数类型    | 描述                                        |
|----------------------------|---------|-------------------------------------------|
| firewall_group             | object  | 一个firewall_group对象。                       |
| admin_state_up             | boolean | 防火墙的管理状态，为up (true)或down (false)。默认是true。 |
| description                | String  | 防火墙的描述。                                   |
| id                         | String  | 防火墙ID。                                    |
| egress_firewall_policy_id  | String  | 防火墙的egress策略ID。                           |
| ingress_firewall_policy_id | String  | 防火墙的ingress策略ID。                          |
| name                       | String  | 防火墙名称。                                    |

| 参数         | 参数类型    | 描述                                                             |
|------------|---------|----------------------------------------------------------------|
| tags       | array   | 资源的标记列表。                                                       |
| ports      | array   | 与防火墙相关联的端口id列表。                                                |
| shared     | boolean | 指示此防火墙是否在所有项目之间共享。                                             |
| status     | String  | 防火墙状态。有效值为ACTIVE、INACTIVE、ERROR、PENDING_UPDATE或PENDING_DELETE。 |
| project_id | String  | 资源所属项目的ID。                                                     |
| tenant_id  | String  | 资源所属项目的ID。                                                     |

## 请求示例

```
GET http://{endpoint}/v2.0/fwaas/firewall_groups/{firewall_group_id}
```

## 正常响应示例

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

## 正常响应代码



200

## 错误码

401, 403, 404

## 创建防火墙

### 功能介绍

创建防火墙

### URI

`POST /v2.0/fwaas/firewall_groups`

### 请求消息

| 参数                         | 参数类型    | 是否必选 | 描述                                        |
|----------------------------|---------|------|-------------------------------------------|
| firewall_group             | object  | 是    | 一个firewall_group对象。                       |
| admin_state_up             | boolean | 否    | 防火墙的管理状态，为up (true)或down (false)。默认是true。 |
| description                | String  | 否    | 防火墙的描述。                                   |
| egress_firewall_policy_id  | String  | 否    | 防火墙的egress策略ID。                           |
| ingress_firewall_policy_id | String  | 否    | 防火墙的ingress策略ID。                          |
| name                       | String  | 否    | 防火墙名称。                                    |
| ports                      | array   | 否    | 与防火墙相关联的端口id列表。                           |
| shared                     | boolean | 否    | 指示此防火墙是否在所有项目之间共享。                        |
| project_id                 | String  | 否    | 资源所属项目的ID。                                |

| 参数        | 参数类型   | 是否必选 | 描述         |
|-----------|--------|------|------------|
| tenant_id | String | 否    | 资源所属项目的ID。 |

## 响应消息

| 参数                         | 参数类型    | 描述                                                             |
|----------------------------|---------|----------------------------------------------------------------|
| firewall_group             | object  | 一个firewall_group对象。                                            |
| admin_state_up             | boolean | 防火墙的管理状态，为up (true)或down (false)。默认是true。                      |
| description                | String  | 防火墙的描述。                                                        |
| id                         | String  | 防火墙ID。                                                         |
| egress_firewall_policy_id  | String  | 防火墙的egress策略ID。                                                |
| ingress_firewall_policy_id | String  | 防火墙的ingress策略ID。                                               |
| name                       | String  | 防火墙名称。                                                         |
| tags                       | array   | 资源的标记列表。                                                       |
| ports                      | array   | 与防火墙相关联的端口id列表。                                                |
| shared                     | boolean | 指示此防火墙是否在所有项目之间共享。                                             |
| status                     | String  | 防火墙状态。有效值为ACTIVE、INACTIVE、ERROR、PENDING_UPDATE或PENDING_DELETE。 |
| project_id                 | String  | 资源所属项目的ID。                                                     |
| tenant_id                  | String  | 资源所属项目的ID。                                                     |

## 请求示例

```
POST http://{endpoint}/v2.0/fwaas/firewall_groups
```

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

## 正常响应示例

```
{
  "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": "",
    "tags": [],
    "ports": [
      "650bfd2f-7766-4a0d-839f-218f33e16998"
    ],
    "project_id": "45977fa2dbd7482098dd68d0d8970117",
    "shared": true,
    "status": "PENDING_CREATE",
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

## 正常响应代码

201

## 错误码

400, 401

## 更新防火墙

## 功能介绍

更新防火墙

## URI

```
PUT /v2.0/fwaas/firewall_groups/{firewall_group_id}
```

| 参数                | 是否必选 | 描述      |
|-------------------|------|---------|
| firewall_group_id | 是    | 防火墙的ID。 |

## 请求消息

| 参数                         | 参数类型    | 是否必选 | 描述                                                            |
|----------------------------|---------|------|---------------------------------------------------------------|
| firewall_group             | object  | 是    | 一个firewall_group对象。                                           |
| admin_state_up             | boolean | 否    | 防火墙的管理状态，为up (true)或down (false)。默认是true。                     |
| description                | String  | 否    | 防火墙的描述。                                                       |
| egress_firewall_policy_id  | String  | 否    | 防火墙的egress策略ID。                                               |
| ingress_firewall_policy_id | String  | 否    | 防火墙的ingress策略ID。                                              |
| name                       | String  | 否    | 防火墙名称。                                                        |
| ports                      | array   | 否    | 与防火墙相关联的虚拟网卡ID列表，所有虚拟网卡的类型必须保持一致（即都是子网网关的虚拟网卡，或都是路由器网关的虚拟网卡）。 |
| shared                     | boolean | 否    | 指示此防火墙是否在所有项目之间共享。                                            |

## 响应消息

| 参数                         | 参数类型    | 描述                                                             |
|----------------------------|---------|----------------------------------------------------------------|
| firewall_group             | object  | 一个firewall_group对象。                                            |
| admin_state_up             | boolean | 防火墙的管理状态，为up (true)或down (false)。默认是true。                      |
| description                | String  | 防火墙的描述。                                                        |
| id                         | String  | 防火墙ID。                                                         |
| egress_firewall_policy_id  | String  | 防火墙的egress策略ID。                                                |
| ingress_firewall_policy_id | String  | 防火墙的ingress策略ID。                                               |
| name                       | String  | 防火墙名称。                                                         |
| tags                       | array   | 资源的标记列表。                                                       |
| ports                      | array   | 与防火墙相关联的端口id列表。                                                |
| shared                     | boolean | 指示此防火墙是否在所有项目之间共享。                                             |
| status                     | String  | 防火墙状态。有效值为ACTIVE、INACTIVE、ERROR、PENDING_UPDATE或PENDING_DELETE。 |
| project_id                 | String  | 资源所属项目的ID。                                                     |
| tenant_id                  | String  | 资源所属项目的ID。                                                     |

## 请求示例

```
PUT http://{endpoint}/v2.0/fwaas/firewall_groups/{firewall_group_id}
```

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

```
}  
}
```

## 正常响应示例

```
{  
  "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": "",  
    "tags": [],  
    "ports": [  
      "650bfd2f-7766-4a0d-839f-218f33e16998"  
    ],  
    "shared": true,  
    "project_id": "45977fa2dbd7482098dd68d0d8970117",  
    "status": "PENDING_UPDATE",  
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"  
  }  
}
```

## 正常响应代码

200

## 错误码

400, 401, 404

## 删除防火墙

## 功能介绍

删除防火墙

## URI

```
DELETE /v2.0/fwaas/firewall_groups/{firewall_group_id}
```

| 参数                | 参数类型   | 是否必选 | 描述     |
|-------------------|--------|------|--------|
| firewall_group_id | string | 是    | 防火墙ID。 |

## 请求消息

无

## 响应消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/fwaas/firewall_groups/{firewall_group_id}
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 404, 409

## 查询防火墙策略列表

## 功能介绍

查询防火墙策略列表

## URI

```
GET /v2.0/fwaas/firewall_policies
```

## 请求消息

| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数                | 参数类型    | 描述                                                       |
|-------------------|---------|----------------------------------------------------------|
| firewall_policies | object  | 一个firewall_policy对象的列表。                                  |
| audited           | boolean | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description       | String  | 防火墙策略的描述。                                                |
| firewall_rules    | array   | 与防火墙策略相关联的防火墙规则id列表。                                     |
| id                | String  | 防火墙策略ID。                                                 |
| name              | String  | 防火墙策略名称。                                                 |
| project_id        | String  | 资源所属项目的ID。                                               |
| shared            | boolean | 指示此防火墙策略是否在所有项目之间共享。                                     |
| tenant_id         | String  | 资源所属项目的ID。                                               |

## 请求示例



```
GET http://{endpoint}/v2.0/fwaas/firewall_policies
```

## 正常响应示例

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

## 正常响应代码

200

## 错误码

401, 403

## 查询防火墙策略详情

### 功能介绍

查询防火墙策略详情

### URI

```
GET /v2.0/fwaas/firewall_policies/{firewall_policy_id}
```

| 参数                 | 参数类型   | 是否必选 | 描述       |
|--------------------|--------|------|----------|
| firewall_policy_id | string | 是    | 防火墙策略id。 |

## 请求消息

无

## 响应消息

| 参数             | 参数类型    | 描述                                                       |
|----------------|---------|----------------------------------------------------------|
| audited        | boolean | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description    | String  | 防火墙策略的描述。                                                |
| firewall_rules | array   | 与防火墙策略相关联的防火墙规则id列表。                                     |
| id             | String  | 防火墙策略ID。                                                 |
| name           | String  | 防火墙策略名称。                                                 |
| project_id     | String  | 资源所属项目的ID。                                               |
| shared         | boolean | 指示此防火墙策略是否在所有项目之间共享。                                     |
| tenant_id      | String  | 资源所属项目的ID。                                               |

## 请求示例

```
GET http://{endpoint}/v2.0/fwaas/firewall_policies/{firewall_policy_id}
```

## 正常响应示例

```
{
  "firewall_policy": {
    "audited": false,
```

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

## 正常响应代码

200

## 错误码

401, 403, 404

## 创建防火墙策略

### 功能介绍

创建防火墙策略，建议每个防火墙只关联一个出向防火墙策略和一个入向防火墙策略即可

### URI

```
POST /v2.0/fwaas/firewall_policies
```

### 请求消息

| 参数              | 参数类型    | 是否必选 | 描述                                                       |
|-----------------|---------|------|----------------------------------------------------------|
| firewall_policy | object  | 是    | 一个firewall_policy对象。                                     |
| audited         | boolean | 否    | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |

| 参数             | 参数类型    | 是否必选 | 描述                   |
|----------------|---------|------|----------------------|
| description    | String  | 否    | 防火墙策略的描述。            |
| firewall_rules | array   | 否    | 与防火墙策略相关联的防火墙规则id列表。 |
| name           | String  | 否    | 防火墙策略名称。             |
| project_id     | String  | 否    | 资源所属项目的ID。           |
| shared         | boolean | 否    | 指示此防火墙策略是否在所有项目之间共享。 |
| tenant_id      | String  | 否    | 资源所属项目的ID。           |

## 响应消息

| 参数              | 参数类型    | 描述                                                       |
|-----------------|---------|----------------------------------------------------------|
| firewall_policy | object  | 一个firewall_policy对象。                                     |
| audited         | boolean | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description     | String  | 防火墙策略的描述。                                                |
| firewall_rules  | array   | 与防火墙策略相关联的防火墙规则id列表。                                     |
| id              | String  | 防火墙策略ID。                                                 |
| name            | String  | 防火墙策略名称。                                                 |
| project_id      | String  | 资源所属项目的ID。                                               |
| shared          | boolean | 指示此防火墙策略是否在所有项目之间共享。                                     |
| tenant_id       | String  | 资源所属项目的ID。                                               |

## 请求示例

```
POST http://{endpoint}/v2.0/fwaas/firewall_policies
```

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

## 正常响应示例

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

## 正常响应代码

201

## 错误码

400, 401

## 更新防火墙策略

## 功能介绍

更新防火墙策略

## URI

```
PUT /v2.0/fwaas/firewall_policies/{firewall_policy_id}
```

| 参数                 | 参数类型   | 是否必选 | 描述       |
|--------------------|--------|------|----------|
| firewall_policy_id | string | 是    | 防火墙策略id。 |

## 请求消息

| 参数              | 参数类型    | 是否必选 | 描述                                                       |
|-----------------|---------|------|----------------------------------------------------------|
| firewall_policy | object  | 是    | 一个firewall_policy对象。                                     |
| audited         | boolean | 否    | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description     | String  | 否    | 防火墙策略的描述。                                                |
| firewall_rules  | array   | 否    | 与防火墙策略相关联的防火墙规则id列表。                                     |
| name            | String  | 否    | 防火墙策略名称。                                                 |
| project_id      | String  | 否    | 资源所属项目的ID。                                               |
| shared          | boolean | 否    | 指示此防火墙策略是否在所有项目之间共享。                                     |
| tenant_id       | String  | 否    | 资源所属项目的ID。                                               |

## 响应消息

| 参数              | 参数类型    | 描述                                                       |
|-----------------|---------|----------------------------------------------------------|
| firewall_policy | object  | 一个firewall_policy对象。                                     |
| audited         | boolean | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description     | String  | 防火墙策略的描述。                                                |
| firewall_rules  | array   | 与防火墙策略相关联的防火墙规则id列表。                                     |
| id              | String  | 防火墙策略ID。                                                 |
| name            | String  | 防火墙策略名称。                                                 |
| project_id      | String  | 资源所属项目的ID。                                               |
| shared          | boolean | 指示此防火墙策略是否在所有项目之间共享。                                     |
| tenant_id       | String  | 资源所属项目的ID。                                               |

## 请求示例

```
PUT http://{endpoint}/v2.0/fwaas/firewall_policies/{firewall_policy_id}
```

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

## 正常响应示例

```
{
  "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",
"shared": false,
"tenant_id": "45977fa2dbd7482098dd68d0d8970117"
}
```

## 正常响应代码

200

## 错误码

400, 401, 404

## 删除防火墙策略

### 功能介绍

删除防火墙策略

### URI

```
DELETE /v2.0/fwaas/firewall_policies/{firewall_policy_id}
```

| 参数                 | 参数类型   | 是否必选 | 描述       |
|--------------------|--------|------|----------|
| firewall_policy_id | string | 是    | 防火墙策略id。 |

### 请求消息

无

### 响应消息



无

## 请求示例

```
DELETE http://{endpoint}/v2.0/fwaas/firewall_policies/{firewall_policy_id}
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 404, 409

## 查询防火墙规则列表

## 功能介绍

查询防火墙规则列表

## URI

```
GET /v2.0/fwaas/firewall_rules
```

## 请求消息

| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数                            | 参数类型    | 描述                                                                                |
|-------------------------------|---------|-----------------------------------------------------------------------------------|
| firewall_rules                | array   | 与防火墙策略相关联的防火墙规则id列表。                                                              |
| action                        | String  | API对匹配防火墙规则的流量执行的动作。有效值为allow或deny。默认是deny。                                       |
| description                   | String  | 防火墙规则的描述。                                                                         |
| destination_firewall_group_id | String  | 目的防火墙ID。                                                                          |
| destination_ip_address        | String  | 防火墙规则的目的IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| destination_port              | String  | 防火墙规则的目的端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| enabled                       | boolean | 设置为false将在防火墙策略中禁用此规则。便于有选择地关闭规则，而不必将规则与防火墙策略解除关联。取值为true或false。默认是true。          |
| firewall_policy_id            | String  | 防火墙策略ID。                                                                          |
| id                            | String  | 防火墙规则的ID。                                                                         |
| ip_version                    | integer | 防火墙规则的IP协议版本。有效值为4或6。默认为4。                                                        |
| name                          | String  | 防火墙规则名称。                                                                          |
| project_id                    | String  | 资源所属项目的ID。                                                                        |
| protocol                      | String  | 防火墙规则的IP协议。取值包括:icmp、tcp、udp或null。                                                |
| shared                        | boolean | 指示此防火墙规则是否在所有项目之间共享。                                                              |
| source_firewall_group_id      | string  | 源防火墙ID。                                                                           |

| 参数                | 参数类型   | 描述                                                                               |
|-------------------|--------|----------------------------------------------------------------------------------|
| source_ip_address | String | 防火墙规则的源IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| source_port       | String | 防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| tenant_id         | String | 资源所属项目的ID。                                                                       |

## 请求示例

```
GET http://{endpoint}/v2.0/fwaas/firewall_rules
```

## 正常响应示例

```
{
  "firewall_rules": [
    {
      "action": "allow",
      "description": "",
      "destination_firewall_group_id": null,
      "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",
      "shared": false,
      "source_firewall_group_id": null,
      "source_ip_address": null,
      "source_port": null,
      "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
    }
  ]
}
```

```
}  
]  
}
```

## 正常响应代码

200

## 错误码

401, 403

## 查询防火墙规则详情

### 功能介绍

查询防火墙规则详情

### URI

```
GET /v2.0/fwaas/firewall_rules/{firewall_rule_id}
```

| 参数               | 是否必选 | 描述        |
|------------------|------|-----------|
| firewall_rule_id | 是    | 防火墙规则的ID。 |

### 请求消息

无

### 响应消息

| 参数            | 参数类型   | 描述                                          |
|---------------|--------|---------------------------------------------|
| firewall_rule | object | 一个firewall_rule对象。                          |
| action        | String | API对匹配防火墙规则的流量执行的动作。有效值为allow或deny。默认是deny。 |

| 参数                            | 参数类型    | 描述                                                                                |
|-------------------------------|---------|-----------------------------------------------------------------------------------|
| description                   | String  | 防火墙规则的描述。                                                                         |
| destination_firewall_group_id | String  | 目的防火墙ID。                                                                          |
| destination_ip_address        | String  | 防火墙规则的目的IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| destination_port              | String  | 防火墙规则的目的端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| enabled                       | boolean | 设置为false将在防火墙策略中禁用此规则。便于有选择地关闭规则，而不必将规则与防火墙策略解除关联。取值为true或false。默认是true。          |
| firewall_policy_id            | String  | 防火墙策略ID。                                                                          |
| id                            | String  | 防火墙规则的ID。                                                                         |
| ip_version                    | integer | 防火墙规则的IP协议版本。有效值为4或6。默认为4。                                                        |
| name                          | String  | 防火墙规则名称。                                                                          |
| project_id                    | String  | 资源所属项目的ID。                                                                        |
| protocol                      | String  | 防火墙规则的IP协议。取值包括:icmp、tcp、udp或null。                                                |
| shared                        | boolean | 指示此防火墙规则是否在所有项目之间共享。                                                              |
| source_firewall_group_id      | string  | 源防火墙ID。                                                                           |
| source_ip_address             | String  | 防火墙规则的源IPv4或IPv6地址或CIDR。没有默认值。                                                    |
| source_port                   | String  | 防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。  |

| 参数        | 参数类型   | 描述         |
|-----------|--------|------------|
| tenant_id | String | 资源所属项目的ID。 |

## 请求示例

```
GET http://{endpoint}/v2.0/fwaas/firewall_rules/{firewall_rule_id}
```

## 正常响应示例

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_firewall_group_id": null,
    "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",
    "shared": false,
    "source_firewall_group_id": null,
    "source_ip_address": null,
    "source_port": null,
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 创建防火墙规则

### 功能介绍

创建防火墙规则

### URI

```
POST /v2.0/fwaas/firewall_rules
```

### 请求消息

| 参数                            | 参数类型   | 是否必选 | 描述                                                                                              |
|-------------------------------|--------|------|-------------------------------------------------------------------------------------------------|
| firewall_rule                 | object | 是    | 一个firewall_rule防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| 资源所属项目的ID。e对象。                |        |      |                                                                                                 |
| action                        | String | 是    | API对匹配防火墙规则的流量执行的动作。有效值为allow、deny或reject。                                                      |
| description                   | String | 否    | 防火墙规则的描述。                                                                                       |
| destination_firewall_group_id | String | 否    | 目的防火墙ID。                                                                                        |
| destination_ip_address        | String | 否    | 防火墙规则的目的IPv4或IPv6地址或CIDR。没有默认值。                                                                 |

| 参数                       | 参数类型    | 是否必选 | 描述                                                                                |
|--------------------------|---------|------|-----------------------------------------------------------------------------------|
| destination_port         | String  | 否    | 防火墙规则的目的端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| enabled                  | boolean | 否    | 设置为false将在防火墙策略中禁用此规则。便于有选择地关闭规则，而不必将规则与防火墙策略解除关联。取值为true或false。默认是true。          |
| ip_version               | integer | 否    | 防火墙规则的IP协议版本。有效值为4或6。默认为4。                                                        |
| name                     | String  | 否    | 防火墙规则名称。                                                                          |
| project_id               | String  | 否    | 资源所属项目的ID。                                                                        |
| protocol                 | String  | 否    | 防火墙规则的IP协议。取值包括:icmp、tcp、udp或null。                                                |
| shared                   | boolean | 否    | 指示此防火墙规则是否在所有项目之间共享。                                                              |
| source_firewall_group_id | string  | 否    | 源防火墙ID。                                                                           |
| source_ip_address        | String  | 否    | 防火墙规则的源IPv4或IPv6地址或CIDR。没有默认值。                                                    |
| source_port              | String  | 否    | 防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。  |
| tenant_id                | String  | 否    | 资源所属项目的ID。                                                                        |



## 响应消息

| 参数                            | 参数类型    | 描述                                                                                |
|-------------------------------|---------|-----------------------------------------------------------------------------------|
| firewall_rule                 | object  | 一个firewall_rule对象。                                                                |
| action                        | String  | API对匹配防火墙规则的流量执行的动作。有效值为allow或deny。默认是deny。                                       |
| description                   | String  | 防火墙规则的描述。                                                                         |
| destination_firewall_group_id | String  | 目的防火墙ID。                                                                          |
| destination_ip_address        | String  | 防火墙规则的目的IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| destination_port              | String  | 防火墙规则的目的端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| enabled                       | boolean | 设置为false将在防火墙策略中禁用此规则。便于有选择地关闭规则，而不必将规则与防火墙策略解除关联。取值为true或false。默认是true。          |
| firewall_policy_id            | String  | 防火墙策略ID。                                                                          |
| id                            | String  | 防火墙规则的ID。                                                                         |
| ip_version                    | integer | 防火墙规则的IP协议版本。有效值为4或6。默认为4。                                                        |
| name                          | String  | 防火墙规则名称。                                                                          |
| project_id                    | String  | 资源所属项目的ID。                                                                        |
| protocol                      | String  | 防火墙规则的IP协议。取值包括:icmp、tcp、udp或null。                                                |
| shared                        | boolean | 指示此防火墙规则是否在所有项目之间共享。                                                              |
| source_firewall_group_id      | string  | 源防火墙ID。                                                                           |

| 参数                | 参数类型   | 描述                                                                               |
|-------------------|--------|----------------------------------------------------------------------------------|
| source_ip_address | String | 防火墙规则的源IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| source_port       | String | 防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| tenant_id         | String | 资源所属项目的ID。                                                                       |

## 请求示例

```
POST http://{endpoint}/v2.0/fwaas/firewall_rules
```

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

## 正常响应示例

```
{
  "firewall_rule": {
    "action": "deny",
    "description": "",
    "destination_firewall_group_id": null,
    "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,
"shared": false,
"source_firewall_group_id": null,
"source_ip_address": null,
"source_port": null,
"tenant_id": "95573613ec554b4b8df9f2679c64557b"
}
}
```

## 正常响应代码

201

## 错误码

400, 401

## 更新防火墙规则

### 功能介绍

更新防火墙规则

### URI

```
PUT /v2.0/fwaas/firewall_rules/{firewall_rule_id}
```

| 参数               | 是否必选 | 描述        |
|------------------|------|-----------|
| firewall_rule_id | 是    | 防火墙规则的ID。 |

### 请求消息

| 参数 | 参数类型 | 是否必选 | 描述 |
|----|------|------|----|
|----|------|------|----|

| 参数                            | 参数类型    | 是否必选 | 描述                                                                                              |
|-------------------------------|---------|------|-------------------------------------------------------------------------------------------------|
| firewall_rule                 | object  | 是    | 一个firewall_rule防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| 资源所属项目的ID。e对象。                |         |      |                                                                                                 |
| action                        | String  | 否    | API对匹配防火墙规则的流量执行的动作。有效值为allow或deny或reject。                                                      |
| description                   | String  | 否    | 防火墙规则的描述。                                                                                       |
| destination_firewall_group_id | String  | 否    | 目的防火墙ID。                                                                                        |
| destination_ip_address        | String  | 否    | 防火墙规则的目的IPv4或IPv6地址或CIDR。没有默认值。                                                                 |
| destination_port              | String  | 否    | 防火墙规则的目的端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。               |
| enabled                       | boolean | 否    | 设置为false表示在防火墙策略中禁用此规则。方便有选择地关闭规则，而不必将规则与防火墙策略解关联。有效值为true或false。默认是true。                       |
| firewall_policy_id            | string  | 否    | 防火墙策略ID                                                                                         |
| ip_version                    | integer | 否    | 防火墙规则的IP协议版本。有效值为4或6。默认为4。                                                                      |
| name                          | String  | 否    | 防火墙规则名称。                                                                                        |

| 参数                       | 参数类型    | 是否必选 | 描述                                                                               |
|--------------------------|---------|------|----------------------------------------------------------------------------------|
| project_id               | String  | 否    | 资源所属项目的ID。                                                                       |
| protocol                 | String  | 否    | 防火墙规则的IP协议。取值包括:icmp、tcp、udp或null。                                               |
| shared                   | boolean | 否    | 指示此防火墙规则是否在所有项目之间共享。                                                             |
| source_firewall_group_id | string  | 否    | 源防火墙ID。                                                                          |
| source_ip_address        | String  | 否    | 防火墙规则的源IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| source_port              | String  | 否    | 防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| tenant_id                | String  | 否    | 资源所属项目的ID。                                                                       |

## 响应消息

| 参数                            | 参数类型   | 描述                                          |
|-------------------------------|--------|---------------------------------------------|
| firewall_rule                 | object | 一个firewall_rule对象。                          |
| action                        | String | API对匹配防火墙规则的流量执行的动作。有效值为allow或deny。默认是deny。 |
| description                   | String | 防火墙规则的描述。                                   |
| destination_firewall_group_id | String | 目的防火墙ID。                                    |

| 参数                       | 参数类型    | 描述                                                                                |
|--------------------------|---------|-----------------------------------------------------------------------------------|
| destination_ip_address   | String  | 防火墙规则的目的IPv4或IPv6地址或CIDR。没有默认值。                                                   |
| destination_port         | String  | 防火墙规则的目的端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。 |
| enabled                  | boolean | 设置为false将在防火墙策略中禁用此规则。便于有选择地关闭规则，而不必将规则与防火墙策略解除关联。取值为true或false。默认是true。          |
| firewall_policy_id       | String  | 防火墙策略ID。                                                                          |
| id                       | String  | 防火墙规则的ID。                                                                         |
| ip_version               | integer | 防火墙规则的IP协议版本。有效值为4或6。默认为4。                                                        |
| name                     | String  | 防火墙规则名称。                                                                          |
| project_id               | String  | 资源所属项目的ID。                                                                        |
| protocol                 | String  | 防火墙规则的IP协议。取值包括:icmp、tcp、udp或null。                                                |
| shared                   | boolean | 指示此防火墙规则是否在所有项目之间共享。                                                              |
| source_firewall_group_id | string  | 源防火墙ID。                                                                           |
| source_ip_address        | String  | 防火墙规则的源IPv4或IPv6地址或CIDR。没有默认值。                                                    |
| source_port              | String  | 防火墙规则的源端口或端口范围。有效的值可以是端口号，可以是整数形式，也可以是端口范围，格式为A:分隔的范围。对于端口范围，包括端口范围的两端。例如,80:90。  |
| tenant_id                | String  | 资源所属项目的ID。                                                                        |

## 请求示例

```
PUT http://{endpoint}/v2.0/fwaas/firewall_rules/{firewall_rule_id}
```

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

## 正常响应示例

```
{
  "firewall_rule": {
    "action": "allow",
    "description": "",
    "destination_firewall_group_id": null,
    "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",
    "shared": true,
    "source_firewall_group_id": null,
    "source_ip_address": null,
    "source_port": null,
    "tenant_id": "45977fa2dbd7482098dd68d0d8970117"
  }
}
```

## 正常响应代码

200

## 错误码

400, 401, 404

## 插入防火墙规则

### 功能介绍

插入防火墙规则

### URI

```
PUT /v2.0/fwaas/firewall_policies/{firewall_policy_id}/insert_rule
```

| 参数                 | 参数类型   | 是否必选 | 描述       |
|--------------------|--------|------|----------|
| firewall_policy_id | string | 是    | 防火墙策略id。 |

### 请求消息

| 参数               | 参数类型   | 是否必选 | 描述                                                                                                               |
|------------------|--------|------|------------------------------------------------------------------------------------------------------------------|
| firewall_rule_id | string | 是    | 防火墙规则id。                                                                                                         |
| insert_after     | string | 是    | 后面要插入新规则的firewall_rule的ID。新规则将立即插入到指定的firewall_rule之后。如果同时提供了前后值，则后值将被忽略。要将一条规则插入到还没有规则的策略中，before和after值都必须为""。 |
| insert_before    | string | 是    | 插入新规则的firewall_rule的ID。新规则将被插入到指定的firewall_rule之前。如果同时提供了前后值，则后值将被忽略。要将一条规则插入到还没有规则的策略中，before和after值都必须为""。     |

### 响应消息



| 参数             | 参数类型    | 描述                                                       |
|----------------|---------|----------------------------------------------------------|
| audited        | boolean | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description    | String  | 防火墙策略描述。                                                 |
| firewall_rules | array   | 与防火墙策略相关联的防火墙规则id列表。                                     |
| id             | String  | 防火墙策略ID。                                                 |
| name           | String  | 防火墙策略名称。                                                 |
| project_id     | String  | 资源所属项目的ID。                                               |
| shared         | boolean | 设置为true可使该防火墙策略对其他项目可见。默认是false。                         |
| tenant_id      | String  | 资源所属项目的ID。                                               |

## 请求示例

PUT

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

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

## 正常响应示例

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

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

## 正常响应代码

201

## 错误码

400, 401, 404, 409

## 删除防火墙规则

## 功能介绍

删除防火墙规则

## URI

```
DELETE /v2.0/fwaas/firewall_rules/{firewall_rule_id}
```

| 参数               | 参数类型   | 是否必选 | 描述        |
|------------------|--------|------|-----------|
| firewall_rule_id | String | 是    | 防火墙规则的ID。 |

## 请求消息

无

## 响应消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/fwaas/firewall_rules/{firewall_rule_id}
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 404, 409

## 移除防火墙规则

## 功能介绍

移除防火墙规则

## URI

```
PUT /v2.0/fwaas/firewall_policies/{firewall_policy_id}/remove_rule
```

| 参数                 | 参数类型   | 是否必选 | 描述       |
|--------------------|--------|------|----------|
| firewall_policy_id | string | 是    | 防火墙策略id。 |

## 请求消息

| 参数               | 参数类型   | 是否必选 | 描述       |
|------------------|--------|------|----------|
| firewall_rule_id | string | 是    | 防火墙规则id。 |

## 响应消息

| 参数             | 参数类型    | 描述                                                       |
|----------------|---------|----------------------------------------------------------|
| audited        | boolean | 每次修改防火墙策略或相关规则时，API都会将该属性设置为false。要审计策略，请显式地将此属性设置为true。 |
| description    | String  | 防火墙策略描述。                                                 |
| firewall_rules | array   | 与防火墙策略相关联的防火墙规则id列表。                                     |
| id             | String  | 防火墙策略ID。                                                 |
| name           | String  | 防火墙策略名称。                                                 |
| project_id     | String  | 资源所属项目的ID。                                               |
| shared         | boolean | 设置为true可使该防火墙策略对其他项目可见。默认是false。                         |
| tenant_id      | String  | 资源所属项目的ID。                                               |

## 请求示例

```
PUT
https://{endpoint}/v2.0/fwaas/firewall_policies/{firewall_policy_id}/remove_rule
```

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

## 正常响应示例

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

```
"tenant_id": "95573613ec554b4b8df9f2679c64557b"
}
```

## 正常响应代码

200

## 错误码

400, 401, 404

## 添加防火墙标签

### 功能介绍

添加防火墙标签

### URI

```
PUT /v2.0/firewall_groups/{firewall_group_id}/tags/{tag}
```

| 参数                | 参数类型   | 是否必选    | 描述                                           |
|-------------------|--------|---------|----------------------------------------------|
| firewall_group_id | 是      | 防火墙的ID。 |                                              |
| tag               | string | 是       | 防火墙标签，支持"NetworkACL"和"GatewayFirewall"两种tag。 |

### 请求消息

无

### 响应消息

无

### 请求示例

```
PUT https://{endpoint}/v2.0/firewall_groups/{firewall_group_id}/tags/{tag}
```

## 正常响应示例

无

## 正常响应代码

201

## 错误码

401, 404

## 移除防火墙标签

## 功能介绍

移除防火墙标签

## URI

```
DELETE /v2.0/firewall_groups/{firewall_group_id}/tags/{tag}
```

| 参数                | 参数类型   | 是否必选    | 描述                                           |
|-------------------|--------|---------|----------------------------------------------|
| firewall_group_id | 是      | 防火墙的ID。 |                                              |
| tag               | string | 是       | 防火墙标签，支持"NetworkACL"和"GatewayFirewall"两种tag。 |

## 请求消息

无

## 响应消息

无

## 请求示例

```
DELETE
https://{endpoint}/v2.0/firewall_groups/{firewall_group_id}/tags/{tag}
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 404

## 1.11 NAT网关

### NAT

#### 查询NAT网关列表

##### 功能介绍

查询NAT网关列表

##### URI

```
GET /v2.0/nats
```

##### 请求消息

| 参数          | 参数类型   | 是否必选 | 描述                      |
|-------------|--------|------|-------------------------|
| name        | string | 否    | 根据NAT网关名称过滤列表结果。        |
| id          | string | 否    | 根据NAT网关ID过滤列表结果。        |
| description | string | 否    | 根据NAT网关描述过滤列表结果。        |
| enabled     | string | 否    | 根据NAT网关是否启用过滤列表结果。      |
| project_id  | string | 否    | 根据NAT网关所属项目的项目ID过滤列表结果。 |
| router_id   | string | 否    | 根据NAT网关关联的路由器的ID过滤列表结果。 |



| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数              | 参数类型     | 描述            |
|-----------------|----------|---------------|
| nats            | [object] | NAT网关对象列表。    |
| id              | string   | NAT网关ID。      |
| name            | string   | NAT网关名称。      |
| project_id      | string   | NAT网关所属项目ID。  |
| router_id       | string   | NAT网关关联的路由ID。 |
| enabled         | boolean  | NAT网关是否启用。    |
| description     | string   | NAT网关描述。      |
| has_rules       | boolean  | NAT网关中是否包含规则。 |
| revision_number | integer  | 资源的版本号。       |
| created_at      | string   | 创建时间。         |
| updated_at      | string   | 更新时间。         |
| tenant_id       | string   | NAT网关所属项目ID。  |

## 请求示例

```
GET http://{endpoint}/v2.0/nats
```

## 正常响应示例

```
{
  "nats": [
    {
      "id": "b4062829-b3ec-4103-a6d2-175e62f3b2c7",
      "name": "test_nat",
      "project_id": "a2086024e2ba4dd4b3e804fd3829e8af",
      "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
      "enabled": true,
      "description": "Test NAT API",
      "has_rules": true,
      "revision_number": 0,
      "created_at": "2024-04-26T06:19:56Z",
      "updated_at": "2024-04-26T06:19:56Z",
      "tenant_id": "a2086024e2ba4dd4b3e804fd3829e8af"
    }
  ]
}
```

## 正常响应代码

200

## 错误码

401, 403

## 创建NAT网关

## 功能介绍

创建NAT网关

## URI

```
POST /v2.0/nats
```

## 请求消息

| 参数          | 参数类型    | 是否必选 | 描述            |
|-------------|---------|------|---------------|
| nat         | object  | 是    | NAT网关对象。      |
| name        | string  | 否    | NAT名称。        |
| description | string  | 否    | NAT描述。        |
| router_id   | string  | 是    | 与NAT网关关联的路由器。 |
| enabled     | boolean | 否    | NAT是否启用。      |

## 响应消息

| 参数              | 参数类型    | 描述            |
|-----------------|---------|---------------|
| nat             | object  | NAT网关对象。      |
| id              | string  | NAT网关ID。      |
| name            | string  | NAT网关名称。      |
| project_id      | string  | NAT网关所属项目ID。  |
| router_id       | string  | NAT网关关联的路由ID。 |
| enabled         | boolean | NAT网关是否启用。    |
| description     | string  | NAT网关描述。      |
| has_rules       | boolean | NAT网关中是否包含规则。 |
| revision_number | integer | 资源的版本号。       |
| created_at      | string  | 创建时间。         |
| updated_at      | string  | 更新时间。         |
| tenant_id       | string  | NAT网关所属项目ID。  |

## 请求示例

```
POST http://{endpoint}/v2.0/nats
```

```
{
  "nat": {
    "name": "hello",
    "description": "A NAT.",
    "router_id": "0a280e0f-ce93-4f68-b8f7-b819e812a39a",
    "enabled": true
  }
}
```

## 正常响应示例

```
{
  "nat": {
    "id": "f8711ad9-b6ea-4dd6-98b1-b65d3cf6aa89",
    "name": "test_nat",
    "tenant_id": "dff8006950c146caad161dd0ba545764",
    "enabled": true,
    "description": "A NAT.",
    "has_rules": false,
    "router_id": "0a280e0f-ce93-4f68-b8f7-b819e812a39a",
    "created_at": "2023-11-28T06:10:33Z",
    "updated_at": "2023-11-28T06:10:33Z",
    "revision_number": 0,
    "project_id": "dff8006950c146caad161dd0ba545764"
  }
}
```

## 正常响应代码

201

## 错误码

401, 403

## 查询NAT网关详情

### 功能介绍

查询NAT网关详情

### URI

```
GET /v2.0/nats/{nat_id}
```

| 参数     | 参数类型   | 是否必选 | 描述            |
|--------|--------|------|---------------|
| nat_id | string | 是    | 获取详情的NAT网关ID。 |

### 请求消息

| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

### 响应消息

| 参数   | 参数类型   | 描述       |
|------|--------|----------|
| nat  | object | NAT网关对象。 |
| id   | string | NAT网关ID。 |
| name | string | NAT网关名称。 |

| 参数              | 参数类型    | 描述            |
|-----------------|---------|---------------|
| project_id      | string  | NAT网关所属项目ID。  |
| router_id       | string  | NAT网关关联的路由ID。 |
| enabled         | boolean | NAT网关是否启用。    |
| description     | string  | NAT网关描述。      |
| has_rules       | boolean | NAT网关中是否包含规则。 |
| revision_number | integer | 资源的版本号。       |
| created_at      | string  | 创建时间。         |
| updated_at      | string  | 更新时间。         |
| tenant_id       | string  | NAT网关所属项目ID。  |

## 请求示例

```
GET http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf
```

## 正常响应示例

```
{
  "nat": {
    "id": "e7d9b5e1-0431-4eac-802d-f0764c8e5faf",
    "name": "errrrr11",
    "project_id": "31152673f0684baf82d1978b8bb93adc",
    "router_id": "33088652-bf97-41bd-b74a-e62b21e4c643",
    "enabled": true,
    "description": "2222200",
    "has_rules": false,
    "revision_number": 21,
    "created_at": "2023-11-15T09:26:39Z",
    "updated_at": "2023-11-22T03:32:19Z",
    "tenant_id": "31152673f0684baf82d1978b8bb93adc"
  }
}
```

```
}  
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 删除NAT网关

### 功能介绍

删除NAT网关

### URI

```
DELETE /v2.0/nats/{nat_id}
```

| 参数     | 参数类型   | 是否必选 | 描述            |
|--------|--------|------|---------------|
| nat_id | string | 是    | 需要删除的NAT网关ID。 |

### 请求消息

无

### 响应消息

无

### 请求示例

```
DELETE http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf
```

### 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 403, 404

## 更新NAT网关

### 功能介绍

更新NAT网关

### URI

```
PUT /v2.0/nats/{nat_id}
```

| 参数     | 参数类型   | 是否必选 | 描述            |
|--------|--------|------|---------------|
| nat_id | string | 是    | 需要更新的NAT网关ID。 |

### 请求消息

| 参数          | 参数类型    | 是否必选 | 描述         |
|-------------|---------|------|------------|
| name        | string  | 否    | NAT网关名称。   |
| description | string  | 否    | NAT网关描述。   |
| enabled     | boolean | 否    | NAT网关是否启用。 |

### 响应消息



| 参数              | 参数类型    | 描述            |
|-----------------|---------|---------------|
| nat             | object  | NAT网关对象。      |
| id              | string  | NAT网关ID。      |
| name            | string  | NAT网关名称。      |
| project_id      | string  | NAT网关所属项目ID。  |
| router_id       | string  | NAT网关关联的路由ID。 |
| enabled         | boolean | NAT网关是否启用。    |
| description     | string  | NAT网关描述。      |
| has_rules       | boolean | NAT网关中是否包含规则。 |
| revision_number | integer | 资源的版本号。       |
| created_at      | string  | 创建时间。         |
| updated_at      | string  | 更新时间。         |
| tenant_id       | string  | NAT网关所属项目ID。  |

## 请求示例

```
PUT http://{endpoint}/v2.0/nats/cdb28ce6-4d7f-4214-b848-c76239db9e69
```

```
{
  "nat": {
    "name": "test_nat",
    "description": "A NAT.",
    "enabled": true
  }
}
```

## 正常响应示例

```
{
  "nat": {
    "id": "cdb28ce6-4d7f-4214-b848-c76239db9e69",
    "name": "test_nat",
    "project_id": "dff8006950c146caad161dd0ba545764",
    "router_id": "0a280e0f-ce93-4f68-b8f7-b819e812a39a",
    "enabled": true,
    "description": "A NAT.",
    "has_rules": false,
    "revision_number": 4,
    "created_at": "2023-11-28T06:13:19Z",
    "updated_at": "2023-11-28T06:25:49Z",
    "tenant_id": "dff8006950c146caad161dd0ba545764"
  }
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

# SNAT

## 查询SNAT规则列表

### 功能介绍

查询SNAT规则列表

### URI

```
GET /v2.0/nats/{nat_id}/snat_rules
```

| 参数 | 参数类型 | 是否必选 | 描述 |
|----|------|------|----|
|----|------|------|----|

| 参数     | 参数类型   | 是否必选 | 描述       |
|--------|--------|------|----------|
| nat_id | string | 是    | NAT网关ID。 |

## 请求消息

| 参数            | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|---------------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| id            | string | 否    | 根据SNAT规则的ID过滤列表。                                                                                                                        |
| floatingip_id | string | 否    | 根据FloatingIP的ID过滤列表。                                                                                                                    |
| internal_cidr | string | 否    | 根据CIDR过滤列表。                                                                                                                             |
| enabled       | string | 否    | 根据SNAT规则是否启用过滤列表。                                                                                                                       |
| project_id    | string | 否    | 根据SNAT规则所属的项目ID过滤列表。                                                                                                                    |
| description   | string | 否    | 根据SNAT规则的描述过滤列表。                                                                                                                        |
| fields        | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数            | 参数类型        | 描述                          |
|---------------|-------------|-----------------------------|
| snat_rules    | [object]    | [SNAT对象。]                   |
| id            | string      | SNAT规则ID。                   |
| floatingip_id | string!null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。 |

| 参数                  | 参数类型    | 描述                                                   |
|---------------------|---------|------------------------------------------------------|
| internal_cidr       | string  | SNAT规则适用于CIDR地址，CIDR掩码长度越长，优先级越高。                    |
| enabled             | boolean | 资源是否启用。                                              |
| project_id          | string  | 资源所属的项目ID。                                           |
| external_ip_address | string  | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string  | 资源关联的路由ID。                                           |
| description         | string  | 资源的描述。                                               |
| revision_number     | integer | 资源的版本号。                                              |
| created_at          | string  | 创建时间。                                                |
| updated_at          | string  | 更新时间。                                                |
| tenant_id           | string  | 资源所属的项目ID。                                           |

## 请求示例

```
GET http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/snat_rules
```

## 正常响应示例

```
{
  "snat_rules": [
    {
      "id": "59eda16d-8a17-4fe8-a00e-015ff22e65e9",
      "floatingip_id": "d67bc2a5-d64f-423a-8cc0-230a42216aa5",
      "internal_cidr": "192.168.3.0/24",
      "enabled": true,
      "project_id": "31152673f0684baf82d1978b8bb93adc",
    }
  ]
}
```

```
        "description": "w121w33333",
        "revision_number": 11,
        "external_ip_address": "172.16.10.21",
        "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
        "created_at": "2023-11-16T03:05:04Z",
        "updated_at": "2023-11-28T12:28:50Z",
        "tenant_id": "31152673f0684baf82d1978b8bb93adc"
    },
    {
        "id": "660815c2-db00-40eb-acb4-6385029bcfe5",
        "internal_cidr": "192.168.34.0/25",
        "enabled": true,
        "project_id": "dff8006950c146caad161dd0ba545764",
        "description": "A snat rule.",
        "revision_number": 0,
        "external_ip_address": "172.16.10.21",
        "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
        "created_at": "2023-11-28T11:59:49Z",
        "updated_at": "2023-11-28T11:59:49Z",
        "tenant_id": "dff8006950c146caad161dd0ba545764"
    }
]
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 创建SNAT规则

## 功能介绍

创建SNAT规则

## URI

```
POST /v2.0/nats/{nat_id}/snat_rules
```

| 参数     | 参数类型   | 是否必选 | 描述      |
|--------|--------|------|---------|
| nat_id | string | 是    | NAT网关ID |

## 请求消息

| 参数            | 参数类型        | 是否必选 | 描述                          |
|---------------|-------------|------|-----------------------------|
| snat_rule     | object      | 是    | SNAT规则对象。                   |
| floating_ip   | string!null | 是    | 公网IP的ID，为空时代表使用该路由器的外部IP。   |
| internal_cidr | string      | 是    | 此SNAT规则适用的CIDR，掩码长度越高优先级越高。 |
| enabled       | boolean     | 是    | 是否启用该SNAT规则。                |
| description   | string      | 是    | SNAT规则的描述。                  |

## 响应消息

| 参数            | 参数类型        | 描述                                |
|---------------|-------------|-----------------------------------|
| snat_rule     | object      | SNAT规则对象                          |
| id            | string      | SNAT规则ID。                         |
| floatingip_id | string!null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。       |
| internal_cidr | string      | SNAT规则适用于CIDR地址，CIDR掩码长度越长，优先级越高。 |
| enabled       | boolean     | 资源是否启用。                           |
| project_id    | string      | 资源所属的项目ID。                        |

| 参数                  | 参数类型    | 描述                                                   |
|---------------------|---------|------------------------------------------------------|
| external_ip_address | string  | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string  | 资源关联的路由ID。                                           |
| description         | string  | 资源的描述。                                               |
| revision_number     | integer | 资源的版本号。                                              |
| created_at          | string  | 创建时间。                                                |
| updated_at          | string  | 更新时间。                                                |
| tenant_id           | string  | 资源所属的项目ID。                                           |

## 请求示例

```
POST http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/snat_rules
```

```
{
  "snat_rule": {
    "floating_ip": null,
    "internal_cidr": "192.168.34.0/25",
    "enabled": true,
    "description": "A snat rule."
  }
}
```

## 正常响应示例

```
{
  "snat_rule": {
    "id": "660815c2-db00-40eb-acb4-6385029bcfe5",
    "floatingip_id": null,
    "internal_cidr": "192.168.34.0/25",
  }
}
```

```
"enabled": true,
"project_id": "dff8006950c146caad161dd0ba545764",
"external_ip_address": "172.16.10.21",
"description": "A snat rule.",
"router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
"revision_number": 0,
"created_at": "2023-11-28T11:59:49Z",
"updated_at": "2023-11-28T11:59:49Z",
"tenant_id": "dff8006950c146caad161dd0ba545764"
}
```

## 正常响应代码

201

## 错误码

401, 403, 404

## 查询SNAT规则详情

### 功能介绍

查询SNAT规则详情

### URI

```
GET /v2.0/nats/{nat_id}/snat_rules/{snat_rule_id}
```

| 参数           | 参数类型   | 是否必选 | 描述            |
|--------------|--------|------|---------------|
| nat_id       | string | 是    | NAT网关ID       |
| snat_rule_id | string | 是    | 获取详情的SNAT规则ID |

### 请求消息



| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数                  | 参数类型        | 描述                                                   |
|---------------------|-------------|------------------------------------------------------|
| snat_rule           | object      | SNAT对象。                                              |
| id                  | string      | SNAT规则ID。                                            |
| floatingip_id       | string null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。                          |
| internal_cidr       | string      | SNAT规则适用于CIDR地址，CIDR掩码长度越长，优先级越高。                    |
| enabled             | boolean     | 资源是否启用。                                              |
| project_id          | string      | 资源所属的项目ID。                                           |
| external_ip_address | string      | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string      | 资源关联的路由ID。                                           |
| description         | string      | 资源的描述。                                               |
| revision_number     | integer     | 资源的版本号。                                              |
| created_at          | string      | 创建时间。                                                |
| updated_at          | string      | 更新时间。                                                |

| 参数        | 参数类型   | 描述         |
|-----------|--------|------------|
| tenant_id | string | 资源所属的项目ID。 |

## 请求示例

```
GET http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/snat_rules/660815c2-db00-40eb-acb4-6385029bcfe5
```

## 正常响应示例

```
{
  "snat_rule": {
    "id": "660815c2-db00-40eb-acb4-6385029bcfe5",
    "floatingip_id": "ff3b2ad0-ca22-4955-87d6-2d2d27f78568",
    "internal_cidr": "192.168.34.0/25",
    "enabled": true,
    "project_id": "dff8006950c146caad161dd0ba545764",
    "external_ip_address": "172.16.10.21",
    "description": "A snat rule.",
    "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
    "revision_number": 0,
    "created_at": "2023-11-28T11:59:49Z",
    "updated_at": "2023-11-28T11:59:49Z",
    "tenant_id": "dff8006950c146caad161dd0ba545764"
  }
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 删除SNAT规则

## 功能介绍

删除SNAT规则

## URI

```
DELETE /v2.0/nats/{nat_id}/snat_rules/{snat_rule_id}
```

| 参数           | 参数类型   | 是否必选 | 描述             |
|--------------|--------|------|----------------|
| nat_id       | string | 是    | NAT网关ID。       |
| snat_rule_id | string | 是    | 需要删除的SNAT规则ID。 |

## 请求消息

无

## 响应消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/snat_rules/660815c2-db00-40eb-acb4-6385029bcfe5
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 403, 404

## 更新SNAT规则

## 功能介绍

更新SNAT规则

## URI

```
PUT /v2.0/nats/{nat_id}/snat_rules/{snat_rule_id}
```

| 参数           | 参数类型   | 是否必选 | 描述             |
|--------------|--------|------|----------------|
| nat_id       | string | 是    | NAT网关ID。       |
| snat_rule_id | string | 是    | 需要更新的SNAT规则ID。 |

## 请求消息

| 参数          | 参数类型    | 是否必选 | 描述           |
|-------------|---------|------|--------------|
| snat_rule   | object  | 是    | SNAT规则对象。    |
| enabled     | boolean | 否    | 是否启用该SNAT规则。 |
| description | string  | 否    | SNAT规则的描述。   |

## 响应消息

| 参数            | 参数类型        | 描述                                |
|---------------|-------------|-----------------------------------|
| snat_rule     | object      | SNAT对象。                           |
| id            | string      | SNAT规则ID。                         |
| floatingip_id | string!null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。       |
| internal_cidr | string      | SNAT规则适用于CIDR地址，CIDR掩码长度越长，优先级越高。 |
| enabled       | boolean     | 资源是否启用。                           |

| 参数                  | 参数类型    | 描述                                                   |
|---------------------|---------|------------------------------------------------------|
| project_id          | string  | 资源所属的项目ID。                                           |
| external_ip_address | string  | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string  | 资源关联的路由ID。                                           |
| description         | string  | 资源的描述。                                               |
| revision_number     | integer | 资源的版本号。                                              |
| created_at          | string  | 创建时间。                                                |
| updated_at          | string  | 更新时间。                                                |
| tenant_id           | string  | 资源所属的项目ID。                                           |

## 请求示例

```
PUT http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/snat_rules/660815c2-db00-40eb-acb4-6385029bcfe5
```

```
{
  "snat_rule": {
    "enabled": false,
    "description": "disabled"
  }
}
```

## 正常响应示例

```
{
  "snat_rule": {
    "id": "660815c2-db00-40eb-acb4-6385029bcfe5",
    "floatingip_id": "ff3b2ad0-ca22-4955-87d6-2d2d27f78568",
    "internal_cidr": "192.168.34.0/25",
  }
}
```

```
"enabled": false,  
"project_id": "dff8006950c146caad161dd0ba545764",  
"external_ip_address": "172.16.10.21",  
"description": "321",  
"router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",  
"revision_number": 1,  
"created_at": "2023-11-28T11:59:49Z",  
"updated_at": "2023-12-04T03:19:57Z",  
"tenant_id": "dff8006950c146caad161dd0ba545764"  
}  
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

# DNAT

## 查询DNAT规则列表

## 功能介绍

查询DNAT规则列表

## URI

```
GET /v2.0/nats/{nat_id}/dnat_rules
```

| 参数     | 参数类型   | 是否必选 | 描述      |
|--------|--------|------|---------|
| nat_id | string | 是    | NAT网关ID |

## 请求消息

| 参数            | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|---------------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| id            | string | 否    | 根据DNAT规则的ID过滤列表。                                                                                                                        |
| floatingip_id | string | 否    | 根据FloatingIP的ID过滤列表。                                                                                                                    |
| external_port | string | 否    | 根据外部端口过滤列表。                                                                                                                             |
| protocol      | string | 否    | 根据协议过滤列表。当前仅支持TCP/UDP两种协议。                                                                                                              |
| enabled       | string | 否    | 根据DNAT规则是否启用过滤列表。                                                                                                                       |
| project_id    | string | 否    | 根据DNAT规则所属的项目ID过滤列表。                                                                                                                    |
| description   | string | 否    | 根据DNAT规则的描述过滤列表。                                                                                                                        |
| fields        | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

## 响应消息

| 参数                  | 参数类型        | 描述                          |
|---------------------|-------------|-----------------------------|
| dnat_rules          | [object]    | DNAT规则对象列表                  |
| id                  | string      | DNAT规则ID。                   |
| floatingip_id       | string!null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。 |
| external_port       | integer     | DNAT外部端口。                   |
| internal_ip_address | string      | DNAT内部地址。                   |

| 参数                  | 参数类型    | 描述                                                   |
|---------------------|---------|------------------------------------------------------|
| internal_port       | integer | DNAT内部端口。                                            |
| protocol            | string  | DNAT使用的协议，当前仅支持TCP/UDP两种协议。                          |
| enabled             | boolean | 资源是否启用。                                              |
| project_id          | string  | 所属项目ID。                                              |
| external_ip_address | string  | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string  | 资源关联的路由ID。                                           |
| description         | string  | 资源的描述。                                               |
| revision_number     | integer | 资源的版本号。                                              |
| created_at          | string  | 创建时间。                                                |
| updated_at          | string  | 更新时间。                                                |
| tenant_id           | string  | 所属项目ID。                                              |

## 请求示例

```
GET http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/dnat_rules
```

## 正常响应示例

```
{
  "dnat_rules": [
    {
      "id": "3a2f8a65-9bd9-458d-ad9f-3c11562864a8",
      "floatingip_id": "16e2976f-aae4-462b-b302-ec8b6a327d90",
      "external_port": 80,
      "project_id": "31152673f0684baf82d1978b8bb93adc",
      "protocol": "tcp",
    }
  ]
}
```



```
"internal_ip_address": "192.168.188.31",
"internal_port": 80,
"external_ip_address": "172.16.10.21",
"enabled": true,
"description": "A dnat rule.",
"router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
"revision_number": 2,
"created_at": "2023-11-22T02:40:32Z",
"updated_at": "2023-11-29T02:55:44Z",
"tenant_id": "31152673f0684baf82d1978b8bb93adc"
},
{
  "id": "9f1d13fe-cebc-439c-acf6-79f8885c43b9",
  "external_port": 99,
  "project_id": "dff8006950c146caad161dd0ba545764",
  "protocol": "udp",
  "internal_ip_address": "192.168.33.58",
  "internal_port": 99,
  "external_ip_address": "172.16.10.21",
  "enabled": true,
  "description": "A dnat rule.",
  "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
  "revision_number": 0,
  "created_at": "2023-11-29T03:13:31Z",
  "updated_at": "2023-11-29T03:13:31Z",
  "tenant_id": "dff8006950c146caad161dd0ba545764"
}
]
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 创建DNAT规则

## 功能介绍

## 创建DNAT规则

## URI

POST /v2.0/nats/{nat\_id}/dnat\_rules

| 参数     | 参数类型   | 是否必选 | 描述       |
|--------|--------|------|----------|
| nat_id | string | 是    | NAT网关ID。 |

## 请求消息

| 参数                  | 参数类型        | 是否必选 | 描述                          |
|---------------------|-------------|------|-----------------------------|
| dnat_rule           | object      | 是    | DNAT规则对象。                   |
| floatingip_id       | string!null | 是    | 公网IP的ID，为空时代表使用该路由器的外部IP。   |
| external_port       | integer     | 是    | DNAT规则的外部端口。                |
| internal_ip_address | string      | 是    | DNAT规则的内部地址。                |
| internal_port       | integer     | 是    | DNAT规则的内部端口。                |
| protocol            | string      | 是    | DNAT使用的协议，当前仅支持TCP/UDP两种协议。 |
| enabled             | boolean     | 是    | 是否启用该DNAT规则。                |
| description         | string      | 是    | DNAT规则的描述。                  |

## 响应消息

| 参数 | 参数类型 | 描述 |
|----|------|----|
|----|------|----|

| 参数                  | 参数类型        | 描述                                                   |
|---------------------|-------------|------------------------------------------------------|
| dnat_rule           | object      | dnat规则对象                                             |
| id                  | string      | DNAT规则ID。                                            |
| floatingip_id       | string!null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。                          |
| external_port       | integer     | DNAT外部端口。                                            |
| internal_ip_address | string      | DNAT内部地址。                                            |
| internal_port       | integer     | DNAT内部端口。                                            |
| protocol            | string      | DNAT使用的协议，当前仅支持TCP/UDP两种协议。                          |
| enabled             | boolean     | 资源是否启用。                                              |
| project_id          | string      | 所属项目ID。                                              |
| external_ip_address | string      | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string      | 资源关联的路由ID。                                           |
| description         | string      | 资源的描述。                                               |
| revision_number     | integer     | 资源的版本号。                                              |
| created_at          | string      | 创建时间。                                                |
| updated_at          | string      | 更新时间。                                                |
| tenant_id           | string      | 所属项目ID。                                              |

## 请求示例

```
POST http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/dnat_rules
```

```
{
  "dnat_rule": {
    "floatingip_id": "ff3b2ad0-ca22-4955-87d6-2d2d27f78568",
    "external_port": 39,
    "internal_ip_address": "192.168.33.58",
    "internal_port": 66,
    "protocol": "tcp",
    "enabled": true,
    "description": ""
  }
}
```

## 正常响应示例

```
{
  "dnat_rule": {
    "id": "a424ed3a-6d99-4070-b4d8-3d567ed0ef12",
    "floatingip_id": "ff3b2ad0-ca22-4955-87d6-2d2d27f78568",
    "external_port": 39,
    "project_id": "dff8006950c146caad161dd0ba545764",
    "protocol": "tcp",
    "internal_ip_address": "192.168.33.58",
    "internal_port": 66,
    "external_ip_address": "172.16.10.21",
    "enabled": true,
    "description": "",
    "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
    "revision_number": 0,
    "created_at": "2023-11-29T03:03:59Z",
    "updated_at": "2023-11-29T03:03:59Z",
    "tenant_id": "dff8006950c146caad161dd0ba545764"
  }
}
```

## 正常响应代码

201

## 错误码

401, 403, 404

## 查询DNAT规则详情

### 功能介绍

查询DNAT规则详情

### URI

```
GET /v2.0/nats/{nat_id}/dnat_rules/{dnat_rule_id}
```

| 参数           | 参数类型   | 是否必选 | 描述             |
|--------------|--------|------|----------------|
| nat_id       | string | 是    | NAT网关ID。       |
| dnat_rule_id | string | 是    | 需要更新的DNAT规则ID。 |

### 请求消息

| 参数     | 参数类型   | 是否必选 | 描述                                                                                                                                      |
|--------|--------|------|-----------------------------------------------------------------------------------------------------------------------------------------|
| fields | string | 否    | 希望服务器返回的字段。如果不指定字段查询参数，则网络API返回策略设置允许的所有属性。通过使用fields参数，API只返回所请求的属性集。Fields参数可以指定多次。例如，如果在请求URL中指定fields=id&fields=name，则只返回id和name属性。 |

### 响应消息

| 参数        | 参数类型   | 描述       |
|-----------|--------|----------|
| dnat_rule | object | dnat规则对象 |

| 参数                  | 参数类型        | 描述                                                   |
|---------------------|-------------|------------------------------------------------------|
| id                  | string      | DNAT规则ID。                                            |
| floatingip_id       | string null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。                          |
| external_port       | integer     | DNAT外部端口。                                            |
| internal_ip_address | string      | DNAT内部地址。                                            |
| internal_port       | integer     | DNAT内部端口。                                            |
| protocol            | string      | DNAT使用的协议，当前仅支持TCP/UDP两种协议。                          |
| enabled             | boolean     | 资源是否启用。                                              |
| project_id          | string      | 所属项目ID。                                              |
| external_ip_address | string      | 资源绑定的外部IP地址，如果floatingip_id为空则显示路由器网关的外部地址，否则显示公网地址。 |
| router_id           | string      | 资源关联的路由ID。                                           |
| description         | string      | 资源的描述。                                               |
| revision_number     | integer     | 资源的版本号。                                              |
| created_at          | string      | 创建时间。                                                |
| updated_at          | string      | 更新时间。                                                |
| tenant_id           | string      | 所属项目ID。                                              |

## 请求示例

```
GET http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-f0764c8e5faf/dnat_rules/3a2f8a65-9bd9-458d-ad9f-3c11562864a8
```

## 正常响应示例

```
{
  "dnat_rule": {
    "id": "3a2f8a65-9bd9-458d-ad9f-3c11562864a8",
    "floatingip_id": "16e2976f-aae4-462b-b302-ec8b6a327d90",
    "external_port": 80,
    "project_id": "31152673f0684baf82d1978b8bb93adc",
    "protocol": "tcp",
    "internal_ip_address": "192.168.188.31",
    "internal_port": 80,
    "external_ip_address": "172.16.10.21",
    "enabled": true,
    "description": "",
    "router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
    "revision_number": 0,
    "created_at": "2023-11-22T02:40:32Z",
    "updated_at": "2023-11-22T02:40:32Z",
    "tenant_id": "31152673f0684baf82d1978b8bb93adc"
  }
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 删除DNAT规则

### 功能介绍

删除DNAT规则

### URI

```
DELETE /v2.0/nats/{nat_id}/dnat_rules/{dnat_rule_id}
```

| 参数 | 参数类型 | 是否必选 | 描述 |
|----|------|------|----|
|----|------|------|----|

| 参数           | 参数类型   | 是否必选 | 描述             |
|--------------|--------|------|----------------|
| nat_id       | string | 是    | NAT网关ID。       |
| dnat_rule_id | string | 是    | 需要删除的DNAT规则ID。 |

## 请求消息

无

## 响应消息

无

## 请求示例

```
DELETE http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-  
f0764c8e5faf/dnat_rules/3a2f8a65-9bd9-458d-ad9f-3c11562864a8
```

## 正常响应示例

无

## 正常响应代码

204

## 错误码

401, 403, 404

## 更新DNAT规则

## 功能介绍

更新DNAT规则

## URI

```
PUT /v2.0/nats/{nat_id}/dnat_rules/{dnat_rule_id}
```



| 参数           | 参数类型   | 是否必选 | 描述             |
|--------------|--------|------|----------------|
| nat_id       | string | 是    | NAT网关ID。       |
| dnat_rule_id | string | 是    | 需要更新的DNAT规则ID。 |

## 请求消息

| 参数          | 参数类型    | 是否必选 | 描述           |
|-------------|---------|------|--------------|
| dnat_rule   | object  | 是    | DNAT规则对象。    |
| enabled     | boolean | 否    | 是否启用该DNAT规则。 |
| description | string  | 否    | DNAT规则的描述。   |

## 响应消息

| 参数                    | 参数类型        | 描述                          |
|-----------------------|-------------|-----------------------------|
| dnat_rule             | object      | dnat规则对象。                   |
| id                    | string      | DNAT规则ID。                   |
| floatingip_id         | string!null | 公网地址ID，如果为空则代表使用路由器网关的外部地址。 |
| external_port         | integer     | DNAT外部端口。                   |
| internal_ip_addresses | string      | DNAT内部地址。                   |
| internal_port         | integer     | DNAT内部端口。                   |
| protocol              | string      | DNAT使用的协议，当前仅支持TCP/UDP两种协议。 |
| enabled               | boolean     | 资源是否启用。                     |

| 参数              | 参数类型    | 描述      |
|-----------------|---------|---------|
| project_id      | string  | 所属项目ID。 |
| description     | string  | 资源的描述。  |
| revision_number | integer | 资源的版本号。 |
| created_at      | string  | 创建时间。   |
| updated_at      | string  | 更新时间。   |
| tenant_id       | string  | 所属项目ID。 |

## 请求示例

```
PUT http://{endpoint}/v2.0/nats/e7d9b5e1-0431-4eac-802d-
f0764c8e5faf/dnat_rules/3a2f8a65-9bd9-458d-ad9f-3c11562864a8
```

```
{
  "dnat_rule": {
    "enabled": false,
    "description": "A dnat rule."
  }
}
```

## 正常响应示例

```
{
  "dnat_rule": {
    "id": "3a2f8a65-9bd9-458d-ad9f-3c11562864a8",
    "floatingip_id": "16e2976f-aae4-462b-b302-ec8b6a327d90",
    "external_port": 80,
    "project_id": "31152673f0684baf82d1978b8bb93adc",
    "protocol": "tcp",
    "internal_ip_address": "192.168.188.31",
    "internal_port": 80,
    "external_ip_address": "172.16.10.21",
    "enabled": false,
  }
}
```

```
"description": "A dnat rule.",
"router_id": "9335f398-25d6-411c-a3d9-8773d04c37c7",
"revision_number": 2,
"created_at": "2023-11-22T02:40:32Z",
"updated_at": "2023-11-29T02:55:44Z",
"tenant_id": "31152673f0684baf82d1978b8bb93adc"
}
}
```

## 正常响应代码

200

## 错误码

401, 403

## 1.12 公网IP

### 公网IP

#### 列举公网IP

##### 功能介绍

列举符合条件的所有公网IP

##### URI

```
GET /v2.0/floatingips
```

##### 请求消息

| 参数                    | 参数类型   | 是否必选 | Description                                                          |
|-----------------------|--------|------|----------------------------------------------------------------------|
| router_id             | string | 否    | 指定路由器的ID进行查询。                                                        |
| project_id            | string | 否    | 指定项目ID进行查询。                                                          |
| floating_network_k_id | string | 否    | 公网IP地址申请自外部网络的ID。                                                    |
| port_id               | string | 否    | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。 |
| status                | string | 否    | 公网IP的运行状态。                                                           |
| tags                  | string | 否    | 公网IP的标签。                                                             |

##### 响应消息

| 参数 | 参数类型 | 描述 |
|----|------|----|
|----|------|----|

| 参数                  | 参数类型    | 描述                                                                                                  |
|---------------------|---------|-----------------------------------------------------------------------------------------------------|
| floatingips         | array   | 公网IP列表。                                                                                             |
| id                  | string  | 公网IP的ID。                                                                                            |
| tenant_id           | string  | 公网IP所属的用户ID。                                                                                        |
| floating_ip_address | string  | 公网IP的地址。                                                                                            |
| floating_network_id | string  | 公网IP地址申请自外部网络的ID。                                                                                   |
| router_id           | string  | 绑定公网IP所属router的ID。                                                                                  |
| port_id             | string  | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。                                |
| fixed_ip_addresses  | string  | 绑定公网IP的port的IP地址。                                                                                   |
| status              | string  | 公网IP的运行状态。                                                                                          |
| description         | string  | 公网IP的描述。                                                                                            |
| qos_policy_id       | string  | 公网IP绑定的qos策略。                                                                                       |
| port_details        | string  | 绑定公网IP的port的相关信息，包括port的name、network_id、mac_address、admin_state_up、status、device_id、device_owner信息。 |
| tags                | array   | 公网IP的标签。                                                                                            |
| created_at          | string  | 创建资源时的 UTC 日期和时间戳。                                                                                  |
| updated_at          | string  | 最近一次更新资源时的 UTC 日期和时间戳。                                                                              |
| revision_number     | integer | 公网IP的版本。                                                                                            |
| project_id          | string  | 公网IP所属的项目ID。                                                                                        |

## 请求示例

```
GET http://{endpoint}/v2.0/floatingips
```

## 正常响应示例

```
{
  "floatingips": [{
    "id": "7e1d4f82-b14e-43d3-b4ad-6d9751487293",
    "tenant_id": "bbd16f62d534430bbdf8ca96bf02a3",
    "floating_ip_address": "172.16.10.227",
    "floating_network_id": "0d4df3da-de1e-4a5e-9606-9178efd4a42c",
    "router_id": null,
    "port_id": null,
    "fixed_ip_address": null,
    "status": "DOWN",
    "description": "",
    "qos_policy_id": null,
    "port_details": null,
    "tags": [],
    "created_at": "2022-03-30T03:42:04Z",
    "updated_at": "2022-03-30T03:42:04Z",
    "revision_number": 0,
    "project_id": "bbd16f62d534430bbdf8ca96bf02a3"
  }]
}
```

## 正常响应代码

200

## 错误码

401

## 创建公网IP

## 功能介绍

创建一个公网IP

## URI

```
POST /v2.0/floatingips
```

## 请求消息

| 参数                  | 参数类型    | 是否必选    | Description                                                          |
|---------------------|---------|---------|----------------------------------------------------------------------|
| floatingips         | object  | 公网IP对象。 |                                                                      |
| floating_network_id | boolean | 是       | 公网IP地址申请自外部网络的ID。                                                    |
| qos_policy_id       | string  | 否       | 公网IP绑定的qos策略。                                                        |
| description         | string  | 否       | 公网IP的描述。                                                             |
| port_id             | string  | 否       | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。 |
| floating_ip_address | string  | 否       | 公网IP的地址。                                                             |
| fixed_ip_addresses  | string  | 否       | 绑定公网IP的port的IP地址。                                                    |
| project_id          | string  | 否       | 公网IP所属的项目ID。                                                         |

## 响应消息

| 参数          | 参数类型   | 描述           |
|-------------|--------|--------------|
| floatingips | object | 公网IP对象。      |
| id          | string | 公网IP的ID。     |
| tenant_id   | string | 公网IP所属的用户ID。 |

| 参数                  | 参数类型    | 描述                                                                                                  |
|---------------------|---------|-----------------------------------------------------------------------------------------------------|
| floating_ip_address | string  | 公网IP的地址。                                                                                            |
| floating_network_id | string  | 公网IP地址申请自外部网络的ID。                                                                                   |
| router_id           | string  | 绑定公网IP所属router的ID。                                                                                  |
| port_id             | string  | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。                                |
| fixed_ip_addresses  | string  | 绑定公网IP的port的IP地址。                                                                                   |
| status              | string  | 公网IP的运行状态。                                                                                          |
| description         | string  | 公网IP的描述。                                                                                            |
| qos_policy_id       | string  | 公网IP绑定的qos策略。                                                                                       |
| port_details        | string  | 绑定公网IP的port的相关信息，包括port的name、network_id、mac_address、admin_state_up、status、device_id、device_owner信息。 |
| tags                | array   | 公网IP的标签。                                                                                            |
| created_at          | string  | 创建资源时的 UTC 日期和时间戳。                                                                                  |
| updated_at          | string  | 最近一次更新资源时的 UTC 日期和时间戳。                                                                              |
| revision_number     | integer | 公网IP的版本。                                                                                            |
| project_id          | string  | 公网IP所属的项目ID。                                                                                        |

## 请求示例

```
POST http://{endpoint}/v2.0/floatingips
```



```
{
  "floatingip": {
    "floating_network_id": "0d4df3da-de1e-4a5e-9606-9178efd4a42c"
  }
}
```

## 正常响应示例

```
{
  "floatingip": {
    "id": "7e1d4f82-b14e-43d3-b4ad-6d9751487293",
    "tenant_id": "bbd16f62d534430bbdf8ca96bf02a3",
    "floating_ip_address": "172.16.10.227",
    "floating_network_id": "0d4df3da-de1e-4a5e-9606-9178efd4a42c",
    "router_id": null,
    "port_id": null,
    "fixed_ip_address": null,
    "status": "DOWN",
    "project_id": "bbd16f62d534430bbdf8ca96bf02a3",
    "description": "",
    "qos_policy_id": null,
    "port_details": null,
    "tags": [],
    "created_at": "2022-03-30T03:42:04Z",
    "updated_at": "2022-03-30T03:42:04Z",
    "revision_number": 0
  }
}
```

## 正常响应代码

201

## 错误码

400, 401, 404, 409

## 获取公网IP

## 功能介绍

获取一个公网IP的详情

## URI

```
GET /v2.0/floatingips/{floatingip_id}
```

| 参数            | 是否必选 | 描述           |
|---------------|------|--------------|
| floatingip_id | 是    | 待查询的公网IP的ID。 |

## 请求消息

## 响应消息

| 参数                  | 参数类型   | 描述                                                                   |
|---------------------|--------|----------------------------------------------------------------------|
| floatingips         | object | 公网IP对象。                                                              |
| id                  | string | 公网IP的ID。                                                             |
| tenant_id           | string | 公网IP所属的用户ID。                                                         |
| floating_ip_address | string | 公网IP的地址。                                                             |
| floating_network_id | string | 公网IP地址申请自外部网络的ID。                                                    |
| router_id           | string | 绑定公网IP所属router的ID。                                                   |
| port_id             | string | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。 |
| fixed_ip_addresses  | string | 绑定公网IP的port的IP地址。                                                    |
| status              | string | 公网IP的运行状态。                                                           |

| 参数              | 参数类型    | 描述                                                                                                  |
|-----------------|---------|-----------------------------------------------------------------------------------------------------|
| description     | string  | 公网IP的描述。                                                                                            |
| qos_policy_id   | string  | 公网IP绑定的qos策略。                                                                                       |
| port_details    | string  | 绑定公网IP的port的相关信息，包括port的name、network_id、mac_address、admin_state_up、status、device_id、device_owner信息。 |
| tags            | array   | 公网IP的标签。                                                                                            |
| created_at      | string  | 创建资源时的 UTC 日期和时间戳。                                                                                  |
| updated_at      | string  | 最近一次更新资源时的 UTC 日期和时间戳。                                                                              |
| revision_number | integer | 公网IP的版本。                                                                                            |
| project_id      | string  | 公网IP所属的项目ID。                                                                                        |

## 请求示例

```
GET http://{endpoint}/v2.0/floatingips/145c964a-1951-4a34-b62f-441e0d3e2675
```

## 正常响应示例

```
{
  "floatingip": {
    "id": "7e1d4f82-b14e-43d3-b4ad-6d9751487293",
    "tenant_id": "bbd16f62d534430bbdffe8ca96bf02a3",
    "floating_ip_address": "172.16.10.227",
    "floating_network_id": "0d4df3da-de1e-4a5e-9606-9178efd4a42c",
    "router_id": null,
    "port_id": null,
    "fixed_ip_address": null,
    "status": "DOWN",
    "description": "",
    "qos_policy_id": null,
    "port_details": null,
  }
}
```

```
"tags": [],  
"created_at": "2022-03-30T03:42:04Z",  
"updated_at": "2022-03-30T03:42:04Z",  
"revision_number": 0,  
"project_id": "bbd16f62d534430bbdffe8ca96bf02a3"  
}  
}
```

## 正常响应代码

200

## 错误码

401, 403, 404

## 更新公网IP

### 功能介绍

更新一个公网IP

### URI

```
PUT /v2.0/floatingips/{floatingip_id}
```

| 参数            | 是否必选 | 描述          |
|---------------|------|-------------|
| floatingip_id | 是    | 待查询的公网IP的ID |

## 请求消息

| 参数          | 参数类型   | 是否必选    | Description |
|-------------|--------|---------|-------------|
| floatingips | object | 公网IP对象。 |             |

| 参数                 | 参数类型   | 是否必选 | Description                                                          |
|--------------------|--------|------|----------------------------------------------------------------------|
| port_id            | string | 否    | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。 |
| fixed_ip_addresses | string | 否    | 绑定公网IP的port的IP地址。                                                    |
| qos_policy_id      | string | 否    | 公网IP绑定的qos策略。                                                        |
| description        | string | 否    | 公网IP的描述。                                                             |

## 响应消息

| 参数                  | 参数类型   | 描述                                                                   |
|---------------------|--------|----------------------------------------------------------------------|
| floatingips         | object | 公网IP对象。                                                              |
| id                  | string | 公网IP的ID。                                                             |
| tenant_id           | string | 公网IP所属的用户ID。                                                         |
| floating_ip_address | string | 公网IP的地址。                                                             |
| floating_network_id | string | 公网IP地址申请自外部网络的ID。                                                    |
| router_id           | string | 绑定公网IP所属router的ID。                                                   |
| port_id             | string | 绑定公网IP的port的ID。注意：如果公网IP被NAT网关使用，port_id会被默认置为NAT网关关联的路由器网关的port的ID。 |
| fixed_ip_addresses  | string | 绑定公网IP的port的IP地址。                                                    |
| status              | string | 公网IP的运行状态。                                                           |
| description         | string | 公网IP的描述。                                                             |

| 参数              | 参数类型    | 描述                                                                                                  |
|-----------------|---------|-----------------------------------------------------------------------------------------------------|
| qos_policy_id   | string  | 公网IP绑定的qos策略。                                                                                       |
| port_details    | string  | 绑定公网IP的port的相关信息，包括port的name、network_id、mac_address、admin_state_up、status、device_id、device_owner信息。 |
| tags            | array   | 公网IP的标签。                                                                                            |
| created_at      | string  | 创建资源时的 UTC 日期和时间戳。                                                                                  |
| updated_at      | string  | 最近一次更新资源时的 UTC 日期和时间戳。                                                                              |
| revision_number | integer | 公网IP的版本。                                                                                            |
| project_id      | string  | 公网IP所属的项目ID。                                                                                        |

## 请求示例

```
PUT http://{endpoint}/v2.0/floatingips/145c964a-1951-4a34-b62f-441e0d3e2675
```

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

## 正常响应示例

```
{
  "floatingip": {
    "id": "a3fdad57-8f24-41d3-95cf-1f94c7bc04b1",
    "tenant_id": "45d6353dc853402a8b8987feaccb7429",
    "floating_ip_address": "172.16.10.130",
    "floating_network_id": "0d4df3da-de1e-4a5e-9606-9178efd4a42c",
    "router_id": null,
    "port_id": null,
  }
}
```

```
{
  "fixed_ip_address": null,
  "status": "ACTIVE",
  "description": "",
  "qos_policy_id": "87d450e9-72b9-4711-ac8f-6b0a111e2223",
  "port_details": null,
  "tags": [],
  "created_at": "2022-03-30T06:14:42Z",
  "updated_at": "2022-03-30T06:18:17Z",
  "revision_number": 4,
  "project_id": "45d6353dc853402a8b8987feaccb7429"
}
```

## 正常响应代码

200

## 错误码

400, 401, 404, 409, 412

## 删除公网IP

## 功能介绍

删除一个公网IP

## URI

```
DELETE /v2.0/floatingips/{floatingip_id}
```

| 参数            | 是否必选 | 描述           |
|---------------|------|--------------|
| floatingip_id | 是    | 待查询的公网IP的ID。 |

## 请求消息

## 请求示例

```
DELETE http://{endpoint}/v2.0/floatingips/145c964a-1951-4a34-b62f-441e0d3e2675
```

## 正常响应代码

204

## 错误码

401, 404, 412



## 1.13 发布记录

### 01 <2022-05-31>

| 变更类型 | 变更说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 更新   | <ul style="list-style-type: none"> <li>* <a href="#">列举虚拟网卡</a>: 在响应消息列表中, 新增qos_network_policy_id字段。DHCP占用的虚拟网卡的device_owner字段由"network:dhcp"改为"network:distributed"。</li> <li>* <a href="#">列举QoS policy</a>: 在响应消息列表中, 新增is_default字段。</li> <li>* <a href="#">给QoS policy设置带宽</a>: 在请求消息列表中, 新增bandwidth_limit_rule:direction字段。</li> <li>* <a href="#">给QoS policy设置带宽</a>: 在响应消息列表中, 新增direction字段。</li> <li>* <a href="#">更新路由器</a>: 在请求消息列表中, 移除portforwardings字段。</li> <li>* <a href="#">列举路由器</a>: 在响应消息列表中, 移除portforwardings、flavor_id、ha字段。</li> <li>* <a href="#">创建安全组</a>: 在请求消息列表中, 新增stateful字段。</li> <li>* <a href="#">更新安全组</a>: 在请求消息列表中, 新增stateful字段。</li> <li>* <a href="#">列举安全组</a>: 在响应消息列表中, 新增stateful字段。</li> <li>* <a href="#">列举安全组规则</a>: 在响应消息列表中, 新增normalized_cidr字段。</li> <li>* <a href="#">列举公网IP</a>: 在响应消息列表中, 新增port_details字段。</li> <li>* <a href="#">列举网络</a>: 在响应消息列表中, Network的provider:network_type字段取消支持"vxlan"类型, 新增支持"geneve"类型。</li> <li>* <a href="#">创建网络</a>: 在请求消息列表中, Network的provider:network_type字段取消支持"vxlan"类型, 新增支持"geneve"类型。</li> </ul> |

### 02 <2022-10-31>

| 变更类型 | 变更说明                                                                                  |
|------|---------------------------------------------------------------------------------------|
| 更新   | <ul style="list-style-type: none"> <li>* <a href="#">防火墙API</a>: 添加防火墙API。</li> </ul> |

### 03 <2023-11-13>

| 变更类型 | 变更说明                                                                                     |
|------|------------------------------------------------------------------------------------------|
| 新增   | * <a href="#">网卡add_allowed_address_pairs API</a> : 给指定虚拟网卡增加allowed_address_pairs。      |
| 新增   | * <a href="#">网卡remove_allowed_address_pairs API</a> : 给指定虚拟网卡移除allowed_addresses_pairs。 |
| 新增   | * <a href="#">克隆安全组 API</a> : 创建一个新的安全组，克隆指定安全组的所有规则。                                    |
| 新增   | * <a href="#">列举路由器静态路由 API</a> : 列举指定路由器的静态路由列表。                                        |
| 新增   | * <a href="#">新增路由器静态路由 API</a> : 为路由器增加静态路由。                                            |
| 新增   | * <a href="#">更新路由器静态路由 API</a> : 更新路由器一条静态路由的描述。                                        |
| 新增   | * <a href="#">删除路由器静态路由 API</a> : 删除路由器的静态路由。                                            |
| 新增   | * <a href="#">创建辅助私网IP</a> : 支持创建一个辅助私网IP网卡。                                             |
| 新增   | * <a href="#">创建虚拟IP</a> : 支持创建一个虚拟IP网卡。                                                 |

## 04 <2024-04-26>

| 变更类型 | 变更说明                                                  |
|------|-------------------------------------------------------|
| 新增   | * <a href="#">查询NAT网关列表 API</a> : 查询NAT网关列表。          |
| 新增   | * <a href="#">创建NAT网关 API</a> : 创建一个新的NAT网关。          |
| 新增   | * <a href="#">查询NAT网关详情 API</a> : 查询NAT网关的详情。         |
| 新增   | * <a href="#">删除NAT网关 API</a> : 删除一个已有的NAT网关。         |
| 新增   | * <a href="#">更新NAT网关 API</a> : 更新一个已有的NAT网关信息。       |
| 新增   | * <a href="#">查询SNAT规则列表 API</a> : 查询SNAT规则列表。        |
| 新增   | * <a href="#">创建SNAT规则 API</a> : 基于NAT网关创建一个新的SNAT规则。 |

| 变更类型 | 变更说明                                                  |
|------|-------------------------------------------------------|
| 新增   | * <a href="#">查询SNAT规则详情 API</a> : 查询SNAT规则的详情。       |
| 新增   | * <a href="#">删除SNAT规则 API</a> : 删除一个已有的SNAT规则。       |
| 新增   | * <a href="#">更新SNAT规则 API</a> : 更新一个已有的SNAT规则信息。     |
| 新增   | * <a href="#">查询DNAT规则列表 API</a> : 查询DNAT规则列表。        |
| 新增   | * <a href="#">创建DNAT规则 API</a> : 基于NAT网关创建一个新的DNAT规则。 |
| 新增   | * <a href="#">查询DNAT规则详情 API</a> : 查询DNAT规则的详情。       |
| 新增   | * <a href="#">删除DNAT规则 API</a> : 删除一个已有的DNAT规则。       |
| 新增   | * <a href="#">更新DNAT规则 API</a> : 更新一个已有的DNAT规则信息。     |

## 05 <2025-01-15>

| 变更类型 | 变更说明                                                                 |
|------|----------------------------------------------------------------------|
| 更新   | * <a href="#">创建网络</a> : 在创建网络的请求中, 新增igmp_snooping字段, 用于配置二层组播相关功能。 |
| 更新   | * <a href="#">更新网络</a> : 在更新网络的请求中, 新增igmp_snooping字段, 用于更新二层组播相关功能。 |
| 新增   | * <a href="#">创建二层桥接</a> : 创建一个二层桥接网卡。                               |
| 新增   | * <a href="#">列举策略路由</a> : 列举策略路由的列表。                                |
| 新增   | * <a href="#">创建策略路由</a> : 创建一条策略路由。                                 |
| 新增   | * <a href="#">获取策略路由</a> : 获取指定的策略路由。                                |
| 新增   | * <a href="#">更新策略路由</a> : 更新指定的策略路由。                                |
| 新增   | * <a href="#">删除策略路由</a> : 删除指定的策略路由。                                |

## 06 <2025-06-13>

| 变更类型 | 变更说明                                       |
|------|--------------------------------------------|
| 新增   | * <a href="#">获取网络可用IP</a> ：获取指定网络的可用IP详情。 |

**咨询热线：400-100-3070**

北京易捷思达科技发展有限公司：

北京市海淀区西北旺东路10号院东区23号楼华胜天成科研大楼一层东侧120-123

南京分公司：

江苏省南京市雨花台区软件大道168号润和创智中心B栋一楼西101

上海office：

上海黄浦区西藏中路336号华旭大厦22楼2204

成都分公司：

成都市高新区天府五街168号德必天府五街WE602

邮箱：

[contact@easystack.cn](mailto:contact@easystack.cn) (业务咨询)

[partners@easystack.cn](mailto:partners@easystack.cn)(合作伙伴咨询)

[marketing@easystack.cn](mailto:marketing@easystack.cn) (市场合作)

[training@easystack.cn](mailto:training@easystack.cn) (培训咨询)

[hr@easystack.cn](mailto:hr@easystack.cn) (招聘咨询)