NAT Gateway

API Reference

Issue 01

Date 2023-04-03

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Before You Start

1.1 Overview

Welcome to *NAT Gateway API Reference*. The NAT Gateway service provides the network address translation (NAT) function for Elastic Cloud Servers (ECSs) and Bare Metal Servers (BMSs) in a Virtual Private Cloud (VPC). It also supports this function for servers that connect to a VPC through Direct Connect or Virtual Private Network (VPN) in on-premises data centers. It allows these servers to access the Internet using elastic IP addresses (EIPs) or to provide services accessible from the Internet.

This document describes how to use application programming interfaces (APIs) to perform operations on NAT gateways, such as creating or deleting NAT gateways, or adding SNAT rules. For details about all supported operations, see **API Overview**.

If you plan to access NAT gateways through an API, ensure that you are familiar with NAT gateway concepts. For details, see **Service Overview**.

1.2 API Calling

NAT gateway supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see **Calling APIs**.

1.3 Notes and Constraints

- For details about the constraints on using NAT gateways, see **Notes and Constraints**.
- For more details, see the constraints described in each API.

1.4 Concepts

- Account

  An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset

user passwords and grant user permissions. The account is a payment entity, which should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.

- User

  An IAM user is created by an account in IAM to use cloud services. Each IAM user has its own identity credentials (password and access keys).

  API authentication requires information such as the account name, username, and password.

- Region

  Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified into universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

  For details, see **Region and AZ**.

- AZ

  An AZ comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

- Project

  A project corresponds to a region. Default projects are defined. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

- Enterprise project

  Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated.

  For details about enterprise projects and about how to obtain enterprise project IDs, see ***Enterprise Management User Guide***.

1.5 Selecting an API Type

NAT Gateway API can be of v2.0 and v2. v2 APIs are of a later version and will be continuously optimized. You are advised to use v2 APIs.

2 API Overview

NAT Gateway APIs allow you to use all NAT Gateway functions.

API v2

Table 2-1 NAT gateway API v2

Type	Description
NAT Gateway Service	Create, query, update, and delete NAT gateways, including creating a NAT gateway, querying NAT gateways, querying details about a specified NAT gateway, updating a NAT gateway, and deleting a NAT gateway.
SNAT Rules	Create, query, update, and delete SNAT rules, including creating an SNAT rule, querying SNAT rules, querying details about a specified SNAT rule, updating an SNAT rule, and deleting an SNAT rule.
DNAT Rules	Create, query, update, and delete DNAT rules, including creating a DNAT rule, creating DNAT rules in batches, querying DNAT rules, querying details about a specified DNAT rule, updating a DNAT rule, and deleting a DNAT rule.

Table 2-2 NAT gateway API (v2) description

Type	API	Description
NAT Gateway Service	Creating a NAT Gateway	This API is used to create a NAT gateway.
	Querying NAT Gateways	This API is used to query NAT gateways.
	Querying Details About a Specified NAT Gateway	This API is used to query details about a specified NAT gateway.

Type	API	Description
	Updating a NAT Gateway	This API is used to update a NAT gateway.
	Deleting a NAT Gateway	This API is used to delete a NAT gateway.
SNAT Rules	Creating an SNAT Rule	This API is used to create an SNAT rule.
	Querying SNAT Rules	This API is used to query SNAT rules.
	Querying Details About a Specified SNAT Rule	This API is used to query details about a specified SNAT rule.
	Updating an SNAT Rule	This API is used to update an SNAT rule.
	Deleting an SNAT Rule	This API is used to delete an SNAT rule.
DNAT Rules	Creating a DNAT Rule	This API is used to create a DNAT rule.
	Creating DNAT Rules in Batches	This API is used to create DNAT rules in batches.
	Querying DNAT Rules	This API is used to query DNAT rules.
	Querying Details About a Specified DNAT Rule	This API is used to query details about a specified DNAT rule.
	Updating a DNAT Rule	This API is used to update a DNAT rule.
	Deleting a DNAT Rule	This API is used to delete a DNAT rule.

API v2.0

Table 2-3 NAT gateway API v2.0

Type	Description
Tags	Tag NAT gateways. Tags help you manage the NAT gateways. You can query, add, and delete tags.

Table 2-4 NAT gateway API (v2.0) description

Type	API	Description
Tags	Querying NAT Gateways by Tag	This API is used to filter NAT gateways by tag.
	Adding or Deleting NAT Gateway Tags in Batches	This API is used to add or delete tags of a specific NAT gateway in batches.
	Adding a NAT Gateway Tag	This API is used to add tags to a NAT gateway.
	Deleting a NAT Gateway Tag	This API is used to delete NAT gateway tags.
	Querying NAT Gateway Tags	This API is used to query tags of a specified NAT gateway.
	Querying Tags in a Project	This API is used to query all tags of a resource type in a specified region.

Private NAT Gateways

Table 2-5 Private NAT gateway APIs

Type	Description
Private NAT Gateways	Create, query, update, and delete private NAT gateways, including creating a private NAT gateway, querying private NAT gateways, updating a private NAT gateway, deleting a private NAT gateway, and querying details about a specified private NAT gateway.
DNAT Rules	Create, query, update, and delete DNAT rules, including creating a DNAT rule, querying DNAT rules, updating a DNAT rule, deleting a DNAT rule, and querying details about a specified DNAT rule.
SNAT Rules	Create, query, update, and delete SNAT rules, including creating an SNAT rule, querying SNAT rules, updating an SNAT rule, deleting an SNAT rule, and querying details about a specified SNAT rule.
Transit IP Addresses	Assign, query, and release transit IP addresses, including assigning a transit IP address, querying transit IP addresses, releasing a transit IP address, and querying details about a specified transit IP address.

Table 2-6 Descriptions of private NAT gateway APIs

Type	API	Description
Private NAT Gateways	Querying Private NAT Gateways	This API is used to query private NAT gateways.
	Updating a Private NAT Gateway	This API is used to update a private NAT gateway.
	Deleting a Private NAT Gateway	This API is used to delete a private NAT gateway.
	Creating a Private NAT Gateway	This API is used to create a private NAT gateway.
	Querying Details About a Specified Private NAT Gateway	This API is used to query details about a specified private NAT gateway.
DNAT Rules	Querying DNAT Rules	This API is used to query DNAT rules.
	Updating a DNAT Rule	This API is used to update a DNAT rule.
	Creating a DNAT Rule	This API is used to create a DNAT rule.
	Deleting a DNAT Rule	This API is used to delete a DNAT rule.
	Querying Details About a Specified DNAT Rule	This API is used to query details about a specified DNAT rule.
SNAT Rules	Querying SNAT Rules	This API is used to query SNAT rules.
	Querying Details About a Specified SNAT Rule	This API is used to query details about a specified SNAT rule.
	Updating an SNAT Rule	This API is used to update an SNAT rule.
	Creating an SNAT Rule	This API is used to create an SNAT rule.
	Deleting an SNAT Rule	This API is used to delete an SNAT rule.
Transit IP Addresses	Querying Transit IP Addresses	This API is used to query transit IP addresses.
	Releasing a Transit IP Address	This API is used to release a transit IP address.

Type	API	Description
	Assigning a Transit IP Address	This API is used to assign a transit IP address.
	Querying Details About a Specified Transit IP Address.	This API is used to query details about a specified transit IP address.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for **obtaining a user token** as an example to demonstrate how to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

Table 3-1 URI parameter description

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from Regions and Endpoints. For example, the endpoint of IAM in region **Dublin** is **iam.myhuaweicloud.eu**.
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.

Parameter	Description
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of *Parameter name*=*Parameter value*. For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **Dublin** region, obtain the endpoint of IAM (**iam.myhuaweicloud.eu**) for this region and the **resource-path** (**/v3/auth/tokens**) in the URI of the API used to **obtain a user token**. Then, construct the URI as follows:

```
https://iam.myhuaweicloud.eu/v3/auth/tokens
```

NOTE

To simplify the URI display in this document, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server.

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Same as GET except that the server must return only the response header.
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API used to **obtain a user token**, the request method is **POST**. The request is as follows:

```
POST https://iam.myhuaweicloud.eu/v3/auth/tokens
```


Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Common request header fields are as follows.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of *Hostname:Port number*. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Specifies the type (or format) of the message body. The default value **application/json** is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in **Obtaining a Project ID**.	No This field is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cbaa340f9c0f4

Parameter	Description	Mandatory	Example Value
X-Auth-Token	Specifies the user token. It is a response to the API for **obtaining a user token** (This is the only API that does not require authentication). After the request is processed, the value of **X-Subject-Token** in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQcCo...ggg1BBIINPXsidG9rZ

NOTE

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDKs to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added in the request.

For more details, see "Authentication Using AK/SK" in **Authentication**.

The API used to **obtain a user token** does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://iam.myhuaweicloud.eu/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

This part is optional. The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header.

The request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

In the case of the API used to **obtain a user token**, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******** (login password), and *xxxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from Regions and Endpoints.

NOTE

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see **Obtaining a User Token**.

```
POST https://iam.myhuaweicloud.eu/v3/auth/tokens
Content-Type: application/json
```



```
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxxxxxxxxxxx"
            }
        }
    }
}
```

If all data required for the API request is available, you can send the request to call the API through **curl**, **Postman**, or coding. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- Token authentication: Requests are authenticated using tokens.
- AK/SK authentication: Requests are encrypted using AK/SK pairs. AK/SK authentication is recommended because it is more secure than token authentication.

Token Authentication

NOTE

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API. You can obtain a token by calling the **Obtaining User Token** API.

NAT Gateway is a project-level service. When you call the API, set **auth.scope** in the request body to **project**.

```
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",   // IAM user name
```



```
                "password": "********",  // IAM user password
                "domain": {
                    "name": "domainname"  // Name of the account to which the IAM user belongs
                }
            }
        }
    },
    "scope": {
        "project": {
            "name": "xxxxxxxx"    // Project Name
        }
    }
  }
}
```

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
POST https://iam.myhuaweicloud.eu/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK Authentication

NOTE

AK/SK authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token authentication is recommended.

In AK/SK authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key, which is used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In AK/SK authentication, you can use an AK/SK to sign requests based on the signature algorithm or using the signing SDK. For details about how to sign requests and use the signing SDK, see **API Request Signing Guide**.

NOTE

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

3.3 Response

Status Code

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see **Status Codes**.

For example, if status code **201** is returned for calling the API used to **obtain a user token**, the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**.

Figure 3-1 shows the response header fields for the API used to **obtain a user token**. The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-1 Header fields of the response to the request for obtaining a user token

(Optional) Response Body

The body of a response is often returned in structured format as specified in the **Content-Type** header field. The response body transfers content except the response header.

The following is part of the response body for the API used to **obtain a user token**.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "az-01",
......
```

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
    "error_msg": "The format of message is error",
```



```
    "error_code": "AS.0001"
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

4 V2 APIs of Public NAT Gateways

4.1 NAT Gateway Service

4.1.1 Creating a NAT Gateway

Function

This API is used to create a NAT gateway.

URI

POST /v2/{project_id}/nat_gateways

Table 4-1 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Request

Table 4-2 describes the request parameters.

Table 4-2 Request parameter

Parameter	Mandatory	Type	Description
nat_gateway	Yes	Object	Specifies the NAT gateway object. For details, see Table 4-3.

Table 4-3 Description of the **nat_gateway** field

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	No	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.
spec	Yes	String	Specifies the NAT gateway type. The value can be: • **1**: small type, which supports up to 10,000 SNAT connections. • **2**: medium type, which supports up to 50,000 SNAT connections. • **3**: large type, which supports up to 200,000 SNAT connections. • **4**: extra-large type, which supports up to 1,000,000 SNAT connections.
router_id	Yes	String	Specifies the VPC ID.
internal_network_id	Yes	String	Specifies the network ID of the downstream interface (the next hop of the DVR) of the NAT gateway.
enterprise_project_id	No	String	• Specifies the enterprise project ID. When creating a NAT gateway, associate an enterprise project ID with the NAT gateway. The value **0** indicates the default enterprise project. • The value can contain a maximum of 36 characters. It is string **"0"** or in UUID format with hyphens (-).

Response

Table 4-4 lists response parameters.

Table 4-4 Response parameter

Parameter	Type	Description
nat_gateway	Object	Specifies the NAT gateway object. For details, see **Table 4-5**.

Table 4-5 Description of the **nat_gateway** field

Parameter	Type	Description
id	String	Specifies the NAT gateway ID.
tenant_id	String	Specifies the project ID.
name	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.
spec	String	Specifies the NAT gateway type. The value can be: • **1**: small type, which supports up to 10,000 SNAT connections. • **2**: medium type, which supports up to 50,000 SNAT connections. • **3**: large type, which supports up to 200,000 SNAT connections. • **4**: extra-large type, which supports up to 1,000,000 SNAT connections.
router_id	String	Specifies the VPC ID.
internal_network_id	String	Specifies the network ID of the downstream interface (the next hop of the DVR) of the NAT gateway.
status	String	• Specifies the status of the NAT gateway. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.

Parameter	Type	Description
created_at	String	Specifies when the NAT gateway is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
enterprise_project_id	String	• Specifies the enterprise project ID. When creating a NAT gateway, associate an enterprise project ID with the NAT gateway. The value **0** indicates the default enterprise project. • The value can contain a maximum of 36 characters. It is string **"0"** or in UUID format with hyphens (-).

Examples

- Example request

```
POST https://{Endpoint}/v2/27e25061336f4af590faeabeb7fcd9a3/nat_gateways
{
      "nat_gateway": {
      "name": "nat_001",
      "description": "my nat gateway 01",
      "router_id": "d84f345c-80a1-4fa2-a39c-d0d397c3f09a",
      "internal_network_id": "89d66639-aacb-4929-969d-07080b0f9fd9",
      "spec": "1",
      "enterprise_project_id": "0aad99bc-f5f6-4f78-8404-c598d76b0ed2"
   }
}
```

- Example response

```
{
   "nat_gateway": {
      "router_id": "d84f345c-80a1-4fa2-a39c-d0d397c3f09a",
      "status": "PENDING_CREATE",
      "description": "my nat gateway 01",
      "admin_state_up": true,
      "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
      "created_at": "2017-11-18 07:34:32.203044",
      "spec": "1",
      "internal_network_id": "89d66639-aacb-4929-969d-07080b0f9fd9",
      "id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
      "name": "nat_001",
      "enterprise_project_id": "0aad99bc-f5f6-4f78-8404-c598d76b0ed2"
   }
}
```

Status Codes

See **Status Codes**.

4.1.2 Querying NAT Gateways

Function

This API is used to query NAT gateways. Unless otherwise specified, exact match is applied.

URI

GET /v2/{project_id}/nat_gateways

NOTE

You can type the question mark (?) and ampersand (&) at the end of the URI to define multiple search criteria. All optional parameters can be filtered. For details, see the example request.

Table 4-6 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
limit	No	Integer	Specifies the number of records on each page.
id	No	String	Specifies the NAT gateway ID.
tenant_id	No	String	Specifies the project ID.
name	No	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	No	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.
spec	No	String	Specifies the NAT gateway type. The value can be: • **1**: small type, which supports up to 10,000 SNAT connections. • **2**: medium type, which supports up to 50,000 SNAT connections. • **3**: large type, which supports up to 200,000 SNAT connections. • **4**: extra-large type, which supports up to 1,000,000 SNAT connections.
router_id	No	String	Specifies the VPC ID.
internal_network_id	No	String	Specifies the network ID of the downstream interface (the next hop of the DVR) of the NAT gateway.

Parameter	Mandatory	Type	Description
status	No	String	• Specifies the status of the NAT gateway. • For details about all its values, see Table 8-1.
admin_state_up	No	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	No	String	• Specifies when the NAT gateway is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
enterprise_project_id	No	String	• Specifies the enterprise project ID. The value **0** indicates the default enterprise project. • The value can contain a maximum of 36 characters. It is string **"0"** or in UUID format with hyphens (-).

Request

None

Response

Table 4-7 lists response parameters.

Table 4-7 Response parameter

Parameter	Type	Description
nat_gateways	Array(Object)	Specifies the NAT gateway objects. For details, see Table 4-8.

Table 4-8 Description of the **nat_gateway** field

Parameter	Type	Description
id	String	Specifies the NAT gateway ID.
tenant_id	String	Specifies the project ID.

Parameter	Type	Description
name	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.
spec	String	Specifies the NAT gateway type. The value can be: • **1**: small type, which supports up to 10,000 SNAT connections. • **2**: medium type, which supports up to 50,000 SNAT connections. • **3**: large type, which supports up to 200,000 SNAT connections. • **4**: extra-large type, which supports up to 1,000,000 SNAT connections.
router_id	String	Specifies the VPC ID.
internal_network_id	String	Specifies the network ID of the downstream interface (the next hop of the DVR) of the NAT gateway.
status	String	• Specifies the status of the NAT gateway. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	Specifies when the NAT gateway is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
enterprise_project_id	String	• Specifies the enterprise project ID. The value **0** indicates the default enterprise project. • The value can contain a maximum of 36 characters. It is string **"0"** or in UUID format with hyphens (-).

Examples

- Example request
  GET https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/nat_gateways?status=ACTIVE

- Example response

```
{
   "nat_gateways": [
      {
         "router_id": "b1d81744-5165-48b8-916e-e56626feb88f",
         "status": "ACTIVE",
         "description": "",
         "admin_state_up": true,
         "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
         "created_at": "2017-11-15 14:50:39.505112",
         "spec": "2",
         "internal_network_id": "5930796a-6026-4d8b-8790-6c6bfc9f87e8",
         "id": "a253be25-ae7c-4013-978b-3c0785eccd63",
         "name": "wj3",
         "enterprise_project_id": "0aad99bc-f5f6-4f78-8404-c598d76b0ed2"

      },
      {
         "router_id": "305dc52f-13dd-429b-a2d4-444a1039ba0b",
         "status": "ACTIVE",
         "description": "",
         "admin_state_up": true,
         "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
         "created_at": "2017-11-17 07:41:07.538062",
         "spec": "2",
         "internal_network_id": "fc09463b-4ef8-4c7a-93c8-92d9ca6daf9d",
         "id": "e824f1b4-4290-4ebc-8322-cfff370dbd1e",
         "name": "lyl001",
         "enterprise_project_id": "0aad99bc-f5f6-4f78-8404-c598d76b0ed2"
      }
   ]
}
```

Status Codes

See **Status Codes**.

4.1.3 Querying Details About a Specified NAT Gateway

Function

This API is used to query details about a specified NAT gateway.

URI

GET /v2/{project_id}/nat_gateways/{nat_gateway_id}

Table 4-9 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Parameter	Mandatory	Type	Description
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.

Request

None

Response

Table 4-10 lists response parameters.

Table 4-10 Response parameter

Parameter	Type	Description
nat_gateway	Object	Specifies the NAT gateway object. For details, see Table 4-11.

Table 4-11 Description of the **nat_gateway** field

Parameter	Type	Description
id	String	Specifies the NAT gateway ID.
tenant_id	String	Specifies the project ID.
name	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.

Parameter	Type	Description
spec	String	Specifies the NAT gateway type. The value can be: • **1**: small type, which supports up to 10,000 SNAT connections. • **2**: medium type, which supports up to 50,000 SNAT connections. • **3**: large type, which supports up to 200,000 SNAT connections. • **4**: extra-large type, which supports up to 1,000,000 SNAT connections.
router_id	String	Specifies the VPC ID.
internal_network_id	String	Specifies the network ID of the downstream interface (the next hop of the DVR) of the NAT gateway.
status	String	• Specifies the status of the NAT gateway. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	Specifies when the NAT gateway is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
enterprise_project_id	String	• Specifies the enterprise project ID. The value **0** indicates the default enterprise project. • The value can contain a maximum of 36 characters. It is string **"0"** or in UUID format with hyphens (-).

Examples

- Example request
  ```
  GET https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/nat_gateways/a78fb3eb-1654-4710-8742-3fc49d5f04f8
  ```
- Example response
  ```
  {
      "nat_gateway": {
          "router_id": "d84f345c-80a1-4fa2-a39c-d0d397c3f09a",
          "status": "ACTIVE",
          "description": "my nat gateway 01",
          "admin_state_up": true,
  ```



```
        "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
        "created_at": "2017-11-18 07:34:32.203044",
        "spec": "1",
        "internal_network_id": "89d66639-aacb-4929-969d-07080b0f9fd9",
        "id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "name": "nat_001",
        "enterprise_project_id": "0aad99bc-f5f6-4f78-8404-c598d76b0ed2"

    }
}
```


Status Codes

See **Status Codes**.

4.1.4 Updating a NAT Gateway

Function

This API is used to update a NAT gateway.

NOTE

admin_state_up = True & status = "ACTIVE" can be updated. The name, description, and type of a NAT gateway can be updated.

URI

PUT /v2/{project_id}/nat_gateways/{nat_gateway_id}

Table 4-12 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.

Request

Table 4-13 describes the request parameters.

Table 4-13 Request parameter

Parameter	Mandatory	Type	Description
nat_gateway	Yes	Object	Specifies the NAT gateway object. For details, see **Table 4-14**. Mandatory field: None. Only the **name**, **description**, and **spec** fields can be updated. At least one attribute must be specified for the NAT gateway to be updated.

Table 4-14 Description of the **nat_gateway** field

Parameter	Mandatory	Type	Description
name	No	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	No	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.
spec	No	String	Specifies the NAT gateway type. The value can be: • **1**: small type, which supports up to 10,000 SNAT connections. • **2**: medium type, which supports up to 50,000 SNAT connections. • **3**: large type, which supports up to 200,000 SNAT connections. • **4**: extra-large type, which supports up to 1,000,000 SNAT connections.

Response

Table 4-15 lists response parameters.

Table 4-15 Response parameter

Parameter	Type	Description
nat_gateway	Object	Specifies the NAT gateway object. For details, see **Table 4-16**.

Table 4-16 Description of the **nat_gateway** field

Parameter	Type	Description
id	String	Specifies the NAT gateway ID.
tenant_id	String	Specifies the project ID.
name	String	Specifies the NAT gateway name. You can enter up to 64 characters. The name can contain only digits, letters, underscores (_), and hyphens (-).
description	String	Provides supplementary information about the NAT gateway. You can enter up to 255 characters.
spec	String	Specifies the NAT gateway type. The value can be: • **1**: small scale • **2**: medium scale • **3**: large scale • **4**: extra-large scale
router_id	String	Specifies the VPC ID.
internal_network_id	String	Specifies the network ID of the downstream interface (the next hop of the DVR) of the NAT gateway.
status	String	• Specifies the status of the NAT gateway. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	Specifies when the NAT gateway is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.

Parameter	Type	Description
enterprise_project_id	String	• Specifies the enterprise project ID. The value **0** indicates the default enterprise project. • The value can contain a maximum of 36 characters. It is string **"0"** or in UUID format with hyphens (-).

Examples

- Example request

```
PUT https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/nat_gateways/
a78fb3eb-1654-4710-8742-3fc49d5f04f8
  {
    "nat_gateway": {
        "name": "new_name",
        "description": "new description",
        "spec": "1"
    }
}
```

- Example response

```
{
    "nat_gateway": {
        "router_id": "d84f345c-80a1-4fa2-a39c-d0d397c3f09a",
        "status": "ACTIVE",
        "description": "new description",
        "admin_state_up": true,
        "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
        "created_at": "2017-11-18 07:34:32.203044",
        "spec": "1",
        "internal_network_id": "89d66639-aacb-4929-969d-07080b0f9fd9",
        "id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "name": "new_name",
        "enterprise_project_id": "0aad99bc-f5f6-4f78-8404-c598d76b0ed2"
"
    }
}
```

Status Codes

See **Status Codes**.

4.1.5 Deleting a NAT Gateway

Function

This API is used to delete a NAT gateway.

URI

DELETE /v2/{project_id}/nat_gateways/{nat_gateway_id}

Table 4-17 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.

Request

None

Response

None

Examples

- Example request
  DELETE https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/nat_gateways/a78fb3eb-1654-4710-8742-3fc49d5f04f8
- Example response
  None (STATUS CODE 204)

Status Codes

See **Status Codes**.

4.2 SNAT Rules

4.2.1 Creating an SNAT Rule

Function

This API is used to create an SNAT rule.

NOTE

You can create an SNAT rule only when **status** of the NAT gateway is set to **ACTIVE** and **admin_state_up** of the NAT gateway administrator to **True**.

URI

POST /v2/{project_id}/snat_rules

Table 4-18 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Request

Table 4-19 describes the request parameters.

Table 4-19 Request parameter

Parameter	Mandatory	Type	Description
snat_rule	Yes	Object	Specifies the SNAT rule object. For details, see **Table 4-20**.

Table 4-20 Description of the **snat_rule** field

Parameter	Mandatory	Type	Description
nat_gateway_id	Yes	String	Specifies the NAT gateway ID. For details, see **Querying NAT Gateways**.
network_id	No	String	Specifies the network ID used by the SNAT rule. For details, see**Querying Networks**.Configure either **network_id** or **cidr**.
cidr	No	String	Specifies a CIDR block or a host IP address. This parameter and **network_id** are alternative. If the value of **source_type** is **0**, the CIDR block must be a subset of the VPC subnet CIDR block. If the value of **Source_type** is **1**, **cidr** must be a CIDR block of Direct Connect connection.

Parameter	Mandatory	Type	Description
source_type	No	Integer	**0**: Either **network_id** or **cidr** can be specified in a VPC. **1**: Only **cidr** can be specified over a Direct Connect connection. If no value is entered, the default value **0** (VPC) is used.
floating_ip_id	Yes	String	Specifies the EIP ID. Multiple EIP IDs must be separated using commas (,). The maximum length of the ID is 4096 bytes. The number of EIP IDs cannot exceed 20.
description	No	String	Provides supplementary information about the SNAT rule. You can enter up to 255 characters.

Response

Table 4-21 lists response parameters.

Table 4-21 Response parameter

Parameter	Type	Description
snat_rule	Object	Specifies the SNAT rule object. For details, see Table 4-22.

Table 4-22 Description of the **snat_rule** field

Parameter	Type	Description
id	String	Specifies the SNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
network_id	String	Specifies the network ID used by the SNAT rule.
cidr	String	Specifies a subset of the VPC subnet CIDR block or a CIDR block of Direct Connect connection.

Parameter	Type	Description
source_type	Integer	**0**: Either **network_id** or **cidr** can be specified in a VPC. **1**: Only **cidr** can be specified over a Direct Connect connection. If no value is entered, the default value **0** (VPC) is used.
floating_ip_id	String	• Specifies the EIP ID. Multiple EIP IDs must be separated using commas (,). • The maximum length of the ID is 4096 bytes.
description	String	Provides supplementary information about the SNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the SNAT rule. • For details about all its values, see **Resource Status Description**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the SNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
floating_ip_address	String	• Specifies the EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.

Examples

- Example request
  a. Configure parameter **network_id** in a VPC.

```
POST https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/snat_rules
{
    "snat_rule": {
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "network_id": "eaad9cd6-2372-4be1-9535-9bd37210ae7b",
        "source_type":0,
        "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
        "description": "my snat rule 01"
    }
}
```

  b. Configure parameter **cidr** in a VPC.

```
POST /v2/d199ba7e0ba64899b2e81518104b1526/snat_rules
{
```



```
  "snat_rule": {
       "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
       "cidr": "192.168.1.10/32",
       "source_type":0,
       "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
       "description": "my snat rule 01"
   }
}
```

c. Configure parameter **cidr** over a Direct Connect connection.

```
POST https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/snat_rules
 {
    "snat_rule": {
      "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
      "cidr": "172.30.0.0/24",
      "source_type":1,
      "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
      "description": "my snat rule 01"
    }
 }
```

- Example response

a. Response to the request for specifying the **network_id** for a VPC

```
{
   "snat_rule": {
      "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
      "status": "PENDING_CREATE",
      "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
      "admin_state_up": true,
      "network_id": "eaad9cd6-2372-4be1-9535-9bd37210ae7b",
      "description": "",
      "source_type":0,
      "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
      "created_at": "2017-11-18 07:54:21.665430",
      "id": "5b95c675-69c2-4656-ba06-58ff72e1d338",
      "floating_ip_address": "5.21.11.226"
   }
}
```

b. Response to the request for specifying the CIDR block in a VPC

```
{
   "snat_rule": {
      "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
      "status": "PENDING_CREATE",
      "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
      "admin_state_up": true,
      "cidr": "192.168.1.10/32",
      "description": "",
      "source_type":0,
      "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
      "created_at": "2017-11-18 07:54:21.665430",
      "id": "5b95c675-69c2-4656-ba06-58ff72e1d338",
      "floating_ip_address": "5.21.11.226"
   }
}
```

c. Response to the request for specifying the CIDR block in a VPC

```
{
   "snat_rule": {
      "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
      "status": "PENDING_CREATE",
      "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
      "admin_state_up": true,
      "cidr": "172.30.0.0/24",
      "description": "",
      "source_type":1,
      "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
      "created_at": "2017-11-18 07:54:21.665430",
      "id": "5b95c675-69c2-4656-ba06-58ff72e1d338",
      "floating_ip_address": "5.21.11.226"
```



```
  }
}
```

Status Codes

See **Status Codes**.

4.2.2 Querying SNAT Rules

Function

This API is used to query an SNAT rule list.

URI

GET /v2/{project_id}/snat_rules

NOTE

You can type the question mark (?) and ampersand (&) at the end of the URI to define multiple search criteria. All optional parameters can be filtered. For details, see the example request.

Table 4-23 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
limit	No	Integer	Specifies the number of records displayed on each page.
id	No	String	Specifies the SNAT rule ID.
tenant_id	No	String	Specifies the project ID.
nat_gateway_id	No	String	Specifies the NAT gateway ID.
network_id	No	String	Specifies the network ID used by the SNAT rule.
cidr	No	String	Specifies a subset of the VPC subnet CIDR block or a CIDR block of Direct Connect connection.
source_type	No	Integer	**0**: Either **network_id** or **cidr** can be specified in a VPC. **1**: Only **cidr** can be specified over a Direct Connect connection.
floating_ip_id	No	String	• Specifies the EIP ID. • Maximum length: 4,096 characters

Parameter	Mandatory	Type	Description
floating_ip_address	No	String	• Specifies the EIP. • Maximum length: 1,024 characters
status	No	String	• Specifies the status of the SNAT rule. • For details about all its values, see Table 8-1.
admin_state_up	No	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	No	String	Specifies when the SNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.

Request

None

Response

Table 4-24 lists response parameters.

Table 4-24 Response parameter

Parameter	Type	Description
snat_rules	Array(Object)	Specifies the SNAT rule objects. For details, see Table 4-25.

Table 4-25 Description of the **snat_rule** field

Parameter	Type	Description
id	String	Specifies the SNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.

Parameter	Type	Description
network_id	String	Specifies the network ID used by the SNAT rule.
cidr	String	Specifies a subset of the VPC subnet CIDR block or a CIDR block of Direct Connect connection.
source_type	Integer	**0**: Either **network_id** or **cidr** can be specified in a VPC. **1**: Only **cidr** can be specified over a Direct Connect connection. If no value is entered, the default value **0** (VPC) is used.
floating_ip_id	String	• Specifies the EIP ID. Multiple EIP IDs must be separated using commas (,). • The maximum length of the ID is 4096 bytes.
description	String	Provides supplementary information about the SNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the SNAT rule. • For details about all its values, see **Resource Status Description**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	Specifies when the SNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
floating_ip_address	String	• Specifies the EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.
freezed_ip_address	String	• Specifies the frozen EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.

Examples

- Example request
  GET https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/snat_rules?limit=10

- Example response

```
{
    "snat_rules": [
     {
          "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
          "status": "ACTIVE",
          "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
          "admin_state_up": true,
          "network_id": "9a469561-daac-4c94-88f5-39366e5ea193",
          "source_type":0,
          "tenant_id": "d199ba7e0ba64899b2e81518104b1526",
          "created_at": "2017-11-15 15:44:42.595173",
          "id": "79195d50-0271-41f1-bded-4c089b2502ff",
          "floating_ip_address": "5.21.11.242",
          "freezed_ip_address": "",
          "description": "my snat rule 01"
      },
      {
          "floating_ip_id": "6e496fba-abe9-4f5e-9406-2ad8c809ac8c",
          "status": "ACTIVE",
          "nat_gateway_id": "e824f1b4-4290-4ebc-8322-cfff370dbd1e",
          "admin_state_up": true,
          "network_id": "97e89905-f9c8-4ae3-9856-392b0b2fbe7f",
          "source_type":0,
          "tenant_id": "d199ba7e0ba64899b2e81518104b1526",
          "created_at": "2017-11-17 07:43:44.830845",
          "id": "4a1a10d7-0d9f-4846-8cda-24cffeffef5c",
          "floating_ip_address": "5.21.11.142,5.21.11.143",
          "freezed_ip_address": "5.21.11.142",
          "description": "my snat rule 01"
      }
    ]
}
```

Status Codes

See **Status Codes**.

4.2.3 Querying Details About a Specified SNAT Rule

Function

This API is used to query details about a specified SNAT rule.

URI

GET /v2/{project_id}/snat_rules/{snat_rule_id}

Table 4-26 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Parameter	Mandatory	Type	Description
snat_rule_id	Yes	String	Specifies the SNAT rule ID.

Request

None

Response

Table 4-27 lists response parameters.

Table 4-27 Response parameter

Parameter	Type	Description
snat_rule	Object	Specifies the SNAT rule object. For details, see Table 4-28.

Table 4-28 Description of the **snat_rule** field

Parameter	Type	Description
id	String	Specifies the SNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
network_id	String	Specifies the network ID used by the SNAT rule.
cidr	String	Specifies a subset of the VPC subnet CIDR block or a CIDR block of Direct Connect connection.
source_type	Integer	**0**: Either **network_id** or **cidr** can be specified in a VPC. **1**: Only **cidr** can be specified over a Direct Connect connection. If no value is entered, the default value **0** (VPC) is used.

Parameter	Type	Description
floating_ip_id	String	• Specifies the EIP ID. Multiple EIP IDs must be separated using commas (,). • The maximum length of the ID is 4096 bytes.
description	String	Provides supplementary information about the SNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the SNAT rule. • For details about all its values, see **Resource Status Description**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	Specifies when the SNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
floating_ip_address	String	• Specifies the EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.
freezed_ip_address	String	• Specifies the frozen EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.

Examples

- Example request
  ```
  GET https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/snat_rules/5b95c675-69c2-4656-ba06-58ff72e1d338
  ```
- Example response
  ```
  {
      "snat_rule": {
          "floating_ip_id": "bdc10a4c-d81a-41ec-adf7-de857f7c812a",
          "status": "ACTIVE",
          "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
          "admin_state_up": true,
          "network_id": "eaad9cd6-2372-4be1-9535-9bd37210ae7b",
          "source_type":0,
          "tenant_id": "d199ba7e0ba64899b2e81518104b1526",
          "created_at": "2017-11-18 07:54:21.665430",
          "id": "5b95c675-69c2-4656-ba06-58ff72e1d338",
          "floating_ip_address": "5.21.11.226",
          "freezed_ip_address": "",
          "description": "my snat rule 01"
      }
  }
  ```

Status Codes

See **Status Codes**.

4.2.4 Updating an SNAT Rule

Function

This API is used to update an SNAT rule.

NOTE

You can update the EIP or description only when **status** of the SNAT rule is set to **ACTIVE** and **admin_state_up** of the gateway administrator to **True**.

URI

PUT /v2/{project_id}/snat_rules/{snat_rule_id}

Table 4-29 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
snat_rule_id	Yes	String	Specifies the SNAT rule ID.

Request

Table 4-30 describes the request parameters.

Table 4-30 Request parameter

Parameter	Mandatory	Type	Description
snat_rule	Yes	Object	Specifies the SNAT rule object. For details, see **Table 4-31**.

Table 4-31 Description of the **snat_rule** field

Parameter	Mandatory	Type	Description
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.
public_ip_address	No	String	• Specifies the EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes. • The number of EIPs cannot exceed 20.
description	No	String	Provides supplementary information about the SNAT rule. You can enter up to 255 characters.

Response

Table 4-32 lists response parameters.

Table 4-32 Response parameter

Parameter	Type	Description
snat_rule	**NatGatewaySnatRuleResponseBody** object	Specifies the response body of the SNAT rule.

Table 4-33 NatGatewaySnatRuleResponseBody

Parameter	Type	Description
id	String	Specifies the SNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.

Parameter	Type	Description
network_id	String	Specifies the network ID used by the SNAT rule.
cidr	String	Specifies a subset of the VPC subnet CIDR block or a CIDR block of Direct Connect connection.
source_type	Integer	**0**: Either **network_id** or **cidr** can be specified in a VPC. **1**: Only **cidr** can be specified over a Direct Connect connection. If no value is entered, the default value **0** (VPC) is used.
floating_ip_id	String	• Specifies the EIP ID. Multiple EIP IDs must be separated using commas (,). • The maximum length of the ID is 4096 bytes.
description	String	Provides supplementary information about the SNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the SNAT rule. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the SNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
public_ip_address	String	• Specifies the EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.
floating_ip_address	String	• Specifies the EIP. Multiple EIPs must be separated using commas (,). • The maximum length is 1024 bytes.

Examples

- Example request
  ```
  PUT https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/snat_rules/5b95c675-69c2-4656-ba06-58ff72e1d338
  ```



```
{
    "snat_rule": {
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "description": "my snat rule 01",
        "public_ip_address": "10.15.10.11,10.15.10.12"
    }
}
```

- Example response

```
{
    "snat_rule": {
        "floating_ip_id": " bdc10a4c-d81a-41ec-adf7-de857f7c812a,7a094014-9657-463f-972b-e84d56b931a0",
        "status": "PENDING_UPDATE",
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "admin_state_up": true,
        "network_id": "eaad9cd6-2372-4be1-9535-9bd37210ae7b",
        "source_type":0,
        "tenant_id": "27e25061336f4af590faeabeb7fcd9a3",
        "created_at": "2017-11-18 07:54:21.665430",
        "id": "5b95c675-69c2-4656-ba06-58ff72e1d338",
        "public_ip_address": "10.15.10.11,10.15.10.12",        ,
        "floating_ip_address": "10.15.10.11,10.15.10.12",
        "description": "my snat rule 01"
    }
}
```

Status Codes

See **Status Codes**.

4.2.5 Deleting an SNAT Rule

Function

This API is used to delete an SNAT rule.

URI

DELETE /v2/{project_id}/nat_gateways/{nat_gateway_id}/snat_rules/{snat_rule_id}

Table 4-34 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the ID of the NAT gateway to which the SNAT rule belongs.
snat_rule_id	Yes	String	Specifies the SNAT rule ID.

Request

None

Response

None

Examples

- Example request
  DELETE https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/nat_gateways/f4dfea98-874a-46f7-aa2a-fb348d0ceb02/snat_rules/a78fb3eb-1654-4710-8742-3fc49d5f04f8
- Example response
  None (STATUS CODE 204)

Status Codes

See **Status Codes**.

4.3 DNAT Rules

4.3.1 Creating a DNAT Rule

Function

This API is used to create a DNAT rule.

NOTE

You can create a DNAT rule only when **status** of the NAT gateway is set to **ACTIVE** and **admin_state_up** of the NAT gateway administrator to **True**. Either **port_id** or **private_ip** is used each time. If you create a rule that applies to all port types, set **internal_service_port** to **0**, **external_service_port** to **0**, and **protocol** to **ANY**.

URI

POST /v2/{project_id}/dnat_rules

Table 4-35 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Request

Table 4-36 describes the request parameters.

Table 4-36 Request parameter

Parameter	Mandatory	Type	Description
dnat_rule	Yes	Object	Specifies the DNAT rule object. For details, see **Table 4-37**.

Table 4-37 Description of the **dnat_rule** field

Parameter	Mandatory	Type	Description
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.
port_id	No	String	Specifies the port ID of an ECS or a BMS. This parameter and **private_ip** are alternative.
private_ip	No	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection. This parameter and **port_id** are alternative.
internal_service_port	Yes	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems. The value ranges from 0 to 65535.
floating_ip_id	Yes	String	Specifies the EIP ID.
external_service_port	Yes	Integer	Specifies the port for providing external services. The value ranges from 0 to 65535.
protocol	Yes	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	No	String	Provides supplementary information about the DNAT rule.

Parameter	Mandatory	Type	Description
internal_service_port_range	No	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. • The port range is the same as the value of **external _service_port_range**. • The value ranges from 1 to 65535. • Specify two port numbers separated by a single hyphen (-) and no blank spaces in the format, *x-y*, where *x* is lower than *y*.
external_service_port_range	No	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal _service_port_range**. • The value ranges from 1 to 65535. • Specify two port numbers separated by a single hyphen (-) and no blank spaces in the format, *x-y*, where *x* is lower than *y*.

Response

Table 4-38 lists response parameters.

Table 4-38 Response parameter

Parameter	Type	Description
dnat_rule	Object	Specifies the DNAT rule object. For details, see Table 4-39.

Table 4-39 Description of the **dnat_rule** field

Parameter	Type	Description
id	String	Specifies the DNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
port_id	String	Specifies the port ID of an ECS or a BMS. This parameter and **private_ip** are alternative.

Parameter	Type	Description
private_ip	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection.
internal_service_port	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	String	Specifies the EIP ID.
floating_ip_address	String	Specifies the EIP.
external_service_port	Integer	Specifies the port for providing external services.
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	String	Provides supplementary information about the DNAT rule.
status	String	• Specifies the status of the DNAT rule. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the DNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.

Parameter	Type	Description
internal_service_port_range	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. ● The port range is the same as the value of **external_service_port_range**. ● The value ranges from 1 to 65535.
external_service_port_range	String	Specifies the port range used by the floating IP address for providing external services. ● The port range is the same as the value of **internal_service_port_range**. ● The value ranges from 1 to 65535.

Examples

- Example request
  a. Create a rule for a specified port.

```
POST https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules
{
    "dnat_rule": {
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "internal_service_port": 993,
        "protocol": "tcp",
        "external_service_port": 242,
        "description": "my dnat rule 01"
    }
}
```

  b. Create a rule for all ports.

```
POST https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules
{
    "dnat_rule": {
        "floating_ip_id": "Cf99c679-9f41-4dac-8513-9c9228e713e1",
        "nat_gateway_id": "Dda3a125-2406-456c-a11f-598e10578541",
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "any",
        "external_service_port": 0,
        "description": "my dnat rule 01"
    }
}
```

  c. Create a rule based on the specified port range.

```
POST https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules
{
    "dnat_rule": {
        "floating_ip_id": "Cf99c679-9f41-4dac-8513-9c9228e713e1",
        "nat_gateway_id": "Dda3a125-2406-456c-a11f-598e10578541",
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
```



```
        "protocol": "tcp",
        "external_service_port": 0,
        "description": "my dnat rule 01" ,
        "external_service_port_range":"100-200",
        "internal_service_port_range":"100-200"
    }
}
```

- Example response

  a. Create a response for a specified port.

```
{
    "dnat_rule": {
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "status": "ACTIVE",
        "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
        "admin_state_up": true,
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "internal_service_port": 993,
        "protocol": "tcp",
        "tenant_id": "abc",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "external_service_port": 242,
        "floating_ip_address": "5.21.11.226",
        "description": "my dnat rule 01"
    }
}
```

  b. Create a response for all ports.

```
{
    "dnat_rule": {
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "status": "ACTIVE",
        "nat_gateway_id": "dda3a125-2406-456c-a11f-598e10578541",
        "admin_state_up": true,
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "any",
        "tenant_id": "abc",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "external_service_port": 0,
        "floating_ip_address": "5.21.11.227",
        "description": "my dnat rule 01"
    }
}
```

  c. Create a rule based on the specified port range.

```
{
    "dnat_rule": {
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "status": "ACTIVE",
        "nat_gateway_id": "dda3a125-2406-456c-a11f-598e10578541",
        "admin_state_up": true,
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "tcp",
        "tenant_id": "abc",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "external_service_port": 0,
        "floating_ip_address": "5.21.11.227",
        "description": "my dnat rule 01",
        "internal_service_port_range":"100-200",
        "external_service_port_range":"100-200"
    }
}
```


Status Codes

See **Status Codes**.

4.3.2 Creating DNAT Rules in Batches

Function

This API is used to create DNAT rules in batches.

NOTE

You can create DNAT rules in batches only when **status** of the NAT gateway is set to **ACTIVE** and **admin_state_up** of the NAT gateway administrator to **True**. Either **port_id** or **private_ip** is used each time. If you create a rule that applies to all port types, set **internal_service_port** to **0**, **external_service_port** to **0**, and **protocol** to **ANY**.

URI

POST /v2/{project_id}/dnat_rules/batch

Table 4-40 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Request

Table 4-41 describes the request parameters.

Table 4-41 Request parameter

Parameter	Mandatory	Type	Description
dnat_rules	Yes	Array of **dnat_rule** objects	Specifies the DNAT rule objects. For details, see **Table 4-42**.

Table 4-42 Description of the **dnat_rule** field

Parameter	Mandatory	Type	Description
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.
port_id	No	String	Specifies the port ID of an ECS or a BMS. This parameter and **private_ip** are alternative.
private_ip	No	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection. This parameter and **port_id** are alternative.
internal_service_port	Yes	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	Yes	String	Specifies the EIP ID.
external_service_port	Yes	Integer	Specifies the port for providing external services.
protocol	Yes	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	No	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
internal_service_port_range	No	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. • The port range is the same as the value of **external _service_port_range**. • The value ranges from 1 to 65535. • Specify two port numbers separated by a single hyphen (-) and no blank spaces in the format, *x-y*, where *x* is lower than *y*.
external_service_port_range	No	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal _service_port_range**. • The value ranges from 1 to 65535. • Specify two port numbers separated by a single hyphen (-) and no blank spaces in the format, *x-y*, where *x* is lower than *y*.

Response

Table 4-43 lists response parameters.

Table 4-43 Response parameter

Parameter	Type	Description
dnat_rules	Array(dnat_rule)	Specifies the DNAT rule objects. For details, see Table 4-44.

Table 4-44 Description of the **dnat_rule** field

Parameter	Type	Description
id	String	Specifies the DNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
port_id	String	Specifies the port ID of an ECS or a BMS.
private_ip	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection.
internal_service_port	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	String	Specifies the EIP ID.
floating_ip_address	String	Specifies the EIP.
external_service_port	Integer	Specifies the port for providing external services.
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the DNAT rule. • For details about all its values, see Table 8-1.

Parameter	Type	Description
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the DNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
internal_service_port_range	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. • The port range is the same as the value of **external _service_port_range**. • The value ranges from 1 to 65535.
external_service_port_range	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal _service_port_range**. • The value ranges from 1 to 65535.

Examples

- Example request
  a. Creating rules in batches (The first is the rule when **Port Type** is set to **Specific port**, and the second is the rule when **Port Type** is set to **All ports**.)

```
POST https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules/batch
{
    "dnat_rules": [{
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "internal_service_port": 993,
        "protocol": "tcp",
        "external_service_port": 242
    },
    {
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "nat_gateway_id": "dda3a125-2406-456c-a11f-598e10578541",
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "any",
        "external_service_port": 0
    }]
}
```

- Example response
  a. Response to the request for creating rules in batches

```
{
    "dnat_rules": [{
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
```



```
        "status": "ACTIVE",
        "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
        "admin_state_up": true,
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "private_ip": "",
        "internal_service_port": 993,
        "protocol": "tcp",
        "tenant_id": "abc",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "floating_ip_address": "5.21.11.226",
        "external_service_port": 242,
        "description": "my dnat rule 01"
    },
{
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "status": "ACTIVE",
        "nat_gateway_id": "dda3a125-2406-456c-a11f-598e10578541",
        "admin_state_up": true,
        "port_ID": "",
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "any",
        "tenant_id": "abc",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "floating_ip_address": "5.21.11.227",
        "external_service_port": 0,
        "description": "my dnat rule 01"
    }]
}
```


Status Codes

See **Status Codes**.

4.3.3 Querying DNAT Rules

Function

This API is used to query DNAT rules.

URI

GET /v2/{project_id}/dnat_rules

NOTE

You can type the question mark (?) and ampersand (&) at the end of the URI to define multiple search criteria. All optional parameters can be filtered. For details, see the example request.

Table 4-45 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the number of records displayed on each page.
id	No	String	Specifies the DNAT rule ID.
tenant_id	No	String	Specifies the project ID.
nat_gateway_id	No	String	Specifies the NAT gateway ID.
port_id	No	String	Specifies the port ID of an ECS or a BMS.
private_ip	No	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection.
internal_service_port	No	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	No	String	Specifies the EIP ID.
floating_ip_address	No	String	Specifies the EIP.
external_service_port	No	Integer	Specifies the port for providing external services.
protocol	No	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	No	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
status	No	String	• Specifies the status of the DNAT rule. • For details about all its values, see **Resource Status Description**.
admin_state_up	No	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.

Request

None

Response

Table 4-46 lists response parameters.

Table 4-46 Response parameter

Parameter	Type	Description
dnat_rules	Array(Object)	Specifies the DNAT rule objects. For details, see Table 4-47.

Table 4-47 Description of the **dnat_rule** field

Parameter	Type	Description
id	String	Specifies the DNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
port_id	String	Specifies the port ID of an ECS or a BMS.
private_ip	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection.
internal_service_port	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	String	Specifies the EIP ID.
floating_ip_address	String	Specifies the EIP.
external_service_port	Integer	Specifies the port for providing external services.
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.

Parameter	Type	Description
description	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the DNAT rule. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the DNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
internal_service_port_range	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. • The port range is the same as the value of **external_service_port_range**. • The value ranges from 1 to 65535.
external_service_port_range	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal_service_port_range**. • The value ranges from 1 to 65535.

Examples

- Example request
  ```
  GET https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526d/dnat_rules?limit=10
  ```
- Example response
  ```
  {
      "dnat_rules": [
          {
              "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
              "status": "ACTIVE",
              "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
              "admin_state_up": true,
              "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
              "private_ip": "",
              "internal_service_port": 993,
              "protocol": "tcp",
              "tenant_id": "abc",
              "created_at": "2017-11-15 15:44:42.595173",
              "id": "79195d50-0271-41f1-bded-4c089b2502ff",
              "floating_ip_address": "5.21.11.226",
              "external_service_port": 242,
              "description": "my dnat rule 01"
          },
          {
              "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
              "status": "ACTIVE",
              "nat_gateway_id": "dda3a125-2406-456c-a11f-598e10578541",
              "admin_state_up": true,
              "port_id": "",
              "private_ip": "192.168.1.100",
              "internal_service_port": 0,
              "protocol": "any",
              "tenant_id": "abc",
              "created_at": "2017-11-16 15:44:42.595173",
              "id": "89195d50-0271-41f1-bded-4c089b2502ff",
              "floating_ip_address": "5.21.11.227",
              "external_service_port": 0,
              "description": "my dnat rule 01"
          }
      ]
  }
  ```

Status Codes

See **Status Codes**.

4.3.4 Querying Details About a Specified DNAT Rule

Function

This API is used to query details about a specified DNAT rule.

URI

GET /v2/{project_id}/dnat_rules/{dnat_rule_id}

Table 4-48 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
dnat_rule_id	Yes	String	Specifies the DNAT rule ID.

Request

None

Response

Table 4-49 lists response parameters.

Table 4-49 Response parameter

Parameter	Type	Description
dnat_rule	Object	Specifies the DNAT rule object. For details, see Table 4-50.

Table 4-50 Description of the **dnat_rule** field

Parameter	Type	Description
id	String	Specifies the DNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
port_id	String	Specifies the port ID of an ECS or a BMS.
private_ip	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection.
internal_service_port	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.

Parameter	Type	Description
floating_ip_id	String	Specifies the EIP ID.
floating_ip_address	String	Specifies the EIP.
external_service_port	Integer	Specifies the port for providing external services.
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the DNAT rule. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the DNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.

Parameter	Type	Description
internal_service_port_range	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. • The port range is the same as the value of **external_service_port_range**. • The value ranges from 1 to 65535.
external_service_port_range	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal_service_port_range**. • The value ranges from 1 to 65535.

Examples

- Example request
  ```
  GET https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526d/dnat_rules/5b95c675-69c2-4656-ba06-58ff72e1d338
  ```
- Example response
  ```
  {
      "dnat_rules": {
          "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
          "status": "ACTIVE",
          "nat_gateway_id": "cda3a125-2406-456c-a11f-598e10578541",
          "admin_state_up": true,
          "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
          "private_ip": "",
          "internal_service_port": 993,
          "protocol": "tcp",
          "tenant_id": "d199ba7e0ba64899b2e81518104b1526d",
          "created_at": "2017-11-15 15:44:42.595173",
          "id": "5b95c675-69c2-4656-ba06-58ff72e1d338",
          "floating_ip_address": "5.21.11.226",
          "external_service_port": 242,
          "description": "my dnat rule 01"
      }
  }
  ```

Status Codes

See **Status Codes**.

4.3.5 Updating a DNAT Rule

Function

This API is used to update a DNAT rule.

NOTE

You can update the rule only when **status** of the DNAT rule is set to **ACTIVE** and **admin_state_up** of the NAT gateway administrator to **True**. Either **port_id** or **private_ip** is used each time. If you create a rule that applies to all port types, set **internal_service_port** to **0**, **external_service_port** to **0**, and **protocol** to **ANY**.

The fields, including **port_id**, **private_ip**, **internal_service_port**, **external_service_port**, **floating_ip_id**, **protocol**, **internal_service_port_range**, and **external_service_port_range** must be updated together.

URI

PUT /v2/{project_id}/dnat_rules/{dnat_rule_id}

Table 4-51 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
dnat_rule_id	Yes	String	Specifies the DNAT rule ID.

Request

Table 4-53 describes the request parameters.

Table 4-52 Parameter description

Parameter	Mandatory	Type	Description
dnat_rule	Yes	Object	Specifies the DNAT rule object. For details, see Table 4-53.

Table 4-53 Description of the **dnat_rule** field

Parameter	Mandatory	Type	Description
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.
port_id	No	String	Specifies the port ID of an ECS or a BMS. This parameter and **private_ip** are alternative.
private_ip	No	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection. This parameter and **port_id** are alternative.
internal_service_port	No	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	No	String	Specifies the EIP ID.
external_service_port	No	Integer	Specifies the port for providing external services.
protocol	No	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	No	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
internal_service_port_range	No	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. ● The port range is the same as the value of **external_service_port_range**. ● The value ranges from 1 to 65535. ● Specify two port numbers separated by a single hyphen (-) and no blank spaces in the format, *x-y*, where *x* is lower than *y*.

Parameter	Mandatory	Type	Description
external_service_port_range	No	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal_service_port_range**. • The value ranges from 1 to 65535. • Specify two port numbers separated by a single hyphen (-) and no blank spaces in the format, *x-y*, where *x* is lower than *y*.

Response

Table 4-54 lists response parameters.

Table 4-54 Response parameter

Parameter	Type	Description
dnat_rule	NatGatewayDnatRuleResponseBody object	Specifies the response body of the DNAT rule.

Table 4-55 NatGatewayDnatRuleResponseBody

Parameter	Type	Description
id	String	Specifies the DNAT rule ID.
tenant_id	String	Specifies the project ID.
nat_gateway_id	String	Specifies the NAT gateway ID.
port_id	String	Specifies the port ID of an ECS or a BMS.
private_ip	String	Specifies the private IP address of a user, for example, the IP address of a VPC for Direct Connect connection.
internal_service_port	Integer	Specifies the port used by ECSs or BMSs to provide services for external systems.
floating_ip_id	String	Specifies the EIP ID.

Parameter	Type	Description
floating_ip_address	String	Specifies the EIP.
external_service_port	Integer	Specifies the port for providing external services.
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively.
description	String	Provides supplementary information about the DNAT rule. You can enter up to 255 characters.
status	String	• Specifies the status of the DNAT rule. • For details about all its values, see **Table 8-1**.
admin_state_up	Boolean	• Specifies the unfrozen or frozen state. • The value can be: – **true**: indicates the unfrozen state. – **false**: indicates the frozen state.
created_at	String	• Specifies when the DNAT rule is created (UTC time). Its value rounds to 6 decimal places for seconds. The format is yyyy-mm-dd hh:mm:ss.
internal_service_port_range	String	Specifies the port range used by ECSs or BMSs to provide services for external systems. • The port range is the same as the value of **external _service_port_range**. • The value ranges from 1 to 65535.
external_service_port_range	String	Specifies the port range used by the floating IP address for providing external services. • The port range is the same as the value of **internal _service_port_range**. • The value ranges from 1 to 65535.

Examples

- Example request
  a. Update a rule to apply to a specific port.

```
PUT https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules/
79195d50-0271-41f1-bded-4c089b2502ff
{
    "dnat_rule": {
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "internal_service_port": 993,
```



```
        "protocol": "tcp",
        "external_service_port": 242,
        "description": "my dnat rule 01"
    }
}
```

b. Update a rule to apply to all ports.

```
PUT https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules/
79195d50-0271-41f1-bded-4c089b2502ff
{
    "dnat_rule": {
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "any",
        "external_service_port": 0,
       "description": "my dnat rule 01"
    }
}
```

c. Update the port range of a rule.

```
PUT https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules/
79195d50-0271-41f1-bded-4c089b2502ff
{
    "dnat_rule": {
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "tcp",
        "external_service_port": 0,
        "description": "my dnat rule 01" ,
        "external_service_port_range":"100-200",
        "internal_service_port_range":"100-200"
    }
}
```

d. Update the description of a rule.

```
PUT https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526/dnat_rules/
79195d50-0271-41f1-bded-4c089b2502ff
{
    "dnat_rule": {
      "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
      "description": "my dnat rule 01"
    }
}
```

- Example response

a. Response to update a rule that applies to a specific port

```
{
    "dnat_rule": {
        "status": "ACTIVE",
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "admin_state_up": true,
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "internal_service_port": 993,
        "protocol": "tcp",
        "tenant_id": "abc",
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "floating_ip_address": "5.21.11.226",
        "external_service_port": 242,
        "description": "my dnat rule 01"
    }
}
```

b. Response to update a rule that applies to all ports

```
{
    "dnat_rule": {
```



```
        "status": "ACTIVE",
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "admin_state_up": true,
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "any",
        "tenant_id": "abc",
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "floating_ip_address ": "5.21.11.227",
        "external_service_port": 0,
        "description": "my dnat rule 01"
    }
}
```

c. Response to update the port range of a rule

```
{
    "dnat_rule": {
        "status": "ACTIVE",
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "admin_state_up": true,
        "private_ip": "192.168.1.100",
        "internal_service_port": 0,
        "protocol": "tcp",
        "tenant_id": "abc",
        "floating_ip_id": "bf99c679-9f41-4dac-8513-9c9228e713e1",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "floating_ip_address ": "5.21.11.227",
        "external_service_port": 0,
        "description": "my dnat rule 01",
        "internal_service_port_range": "100-200",
        "external_service_port_range": "100-200"
    }
}
```

d. Response to update the description of a rule

```
{
    "dnat_rule": {
        "status": "ACTIVE",
        "nat_gateway_id": "a78fb3eb-1654-4710-8742-3fc49d5f04f8",
        "admin_state_up": true,
        "port_id": "9a469561-daac-4c94-88f5-39366e5ea193",
        "private_ip": "",
        "internal_service_port": 993,
        "protocol": "tcp",
        "tenant_id": "abc",
        "floating_ip_id": "cf99c679-9f41-4dac-8513-9c9228e713e1",
        "created_at": "2017-11-15 15:44:42.595173",
        "id": "79195d50-0271-41f1-bded-4c089b2502ff",
        "floating_ip_address": "5.21.11.226",
        "external_service_port": 242,
        "description": "my dnat rule 01"
    }
}
```

Status Codes

See **Status Codes**.

4.3.6 Deleting a DNAT Rule

Function

This API is used to delete a DNAT rule.

URI

DELETE /v2/{project_id}/nat_gateways/{nat_gateway_id}/dnat_rules/{dnat_rule_id}

Table 4-56 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the ID of the NAT gateway to which the DNAT rule belongs.
dnat_rule_id	Yes	String	Specifies the DNAT rule ID.

Request

None

Response

None

Examples

- Example request
  ```
  DELETE https://{Endpoint}/v2/d199ba7e0ba64899b2e81518104b1526d/nat_gateways/f4dfea98-874a-46f7-aa2a-fb348d0ceb02/dnat_rules/a78fb3eb-1654-4710-8742-3fc49d5f04f8
  ```
- Example response
  ```
  None (STATUS CODE 204)
  ```

Status Codes

See **Status Codes**.

5 API v2.0

5.1 Tags

5.1.1 Querying NAT Gateways by Tag

Function

This API is used to filter NAT gateways by tag.

TMS uses this API to filter and list NAT gateways by tag.

By default, NAT gateways and tags are sorted in descending order of creation time.

URI

- URI format

POST /v2.0/{project_id}/nat_gateways/resource_instances/action

- Parameter description

Table 5-1 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Request

Table 5-2 describes the request parameters.

Table 5-2 Request parameters

Parameter	Mandatory	Type	Description
tags	No	Array<Object>	Specifies the included tags. Each tag contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Each tag key must be unique, and each tag value in a tag must be unique. Resources identified by different keys are in AND relationship, and values in one tag are in OR relationship. If no tag filtering criteria is specified, full data is returned.
tags_any	No	Array<Object>	Specifies any included tags. Each tag contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Each tag key must be unique, and each tag value in a tag must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no tag filtering criteria is specified, full data is returned.
not_tags	No	Array<Object>	Specifies the excluded tags. Each tag contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Each tag key must be unique, and each tag value in a tag must be unique. Resources not identified by different keys are in AND relationship, and values in one tag are in OR relationship. If no tag filtering criteria is specified, full data is returned.

Parameter	Mandatory	Type	Description
not_tags_any	No	Array<Object>	Specifies any excluded tags. Each tag contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Each tag key must be unique, and each tag value in a tag must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no tag filtering criteria is specified, full data is returned.
limit	No	String	Number of records. This parameter is not available when **action** is set to **count**. The default value is **1000** when **action** is set to **filter**. The maximum value is **1000**, and the minimum value is **1**. The value cannot be a negative number.
offset	No	String	Specifies the index position. The query starts from the next piece of data indexed by this parameter. This parameter is not required when you query data on the first page. The value in the response returned for querying data on the previous page will be included in this parameter for querying data on subsequent pages. This parameter is not available when **action** is set to **count**. If **action** is set to **filter**, the value must be a number, and the default value is **0**. The value cannot be a negative number.
action	Yes	String	Specifies the operation to perform, which can only be **filter** (filtering) or **count** (querying the total number). The value **filter** indicates pagination query. Value **count** indicates that the total number of query results meeting the search criteria will be returned. Returning other fields is not allowed.

Parameter	Mandatory	Type	Description
matches	No	Array<match>	Specifies the search criteria. The tag key is the field to match, for example, **resource_name**. The tag value indicates the matched value. This field is a fixed dictionary value. Determine whether fuzzy match is required based on different fields. For example, if **key** is **resource_name**, fuzzy search (case insensitive) is used by default. If **value** is an empty string, exact match is used. If **key** is **resource_id**, exact match is used.

Table 5-3 Parameter description of field **tag**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. It can contain a maximum of 127 Unicode characters. **key** cannot be left blank, be an empty string, or be spaces. Before using key, delete spaces before and after the value.
values	Yes	Array<String>	Specifies the tag values. Each value contains a maximum of 255 Unicode characters and cannot contain spaces. Before verification, delete spaces before and after the value. The asterisk (*) is a reserved character. The value can be empty but cannot be left blank. If the value starts with an asterisk (*), the string following the asterisk is fuzzy matched. If the values are null, it indicates **any_value** (querying any value). The values are in OR relationship.

Table 5-4 Description of field **match**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. The value is fixed at **resource_name**.
value	Yes	String	Specifies the key value. It can contain a maximum of 255 Unicode characters.

Response

Table 5-5 describes the response parameters.

Table 5-5 Response parameter

Parameter	Type	Description
resources	Array<resource>	Specifies the resource object list. For details, see **Table 5-6**.
total_count	Integer	Specifies the total number of resources.

Table 5-6 Data structure description of field **resource**

Parameter	Type	Description
resource_id	String	Specifies the resource ID.
resource_detail	Object	Provides details about the resource. The value is a resource object, used for extension. This parameter is left blank by default.
tags	Array<resource_tag>	Specifies the list of queried tags. If no tag is matched, an empty array is returned. For details, see **Table 5-7**.
resource_name	String	Specifies the resource name. This parameter is an empty string by default if there is no resource name.

Table 5-7 Parameter description of field **resource_tag**

Parameter	Type	Description
key	String	Specifies the tag key. It contains a maximum of 36 Unicode characters. A tag key cannot be left blank. It cannot contain non-printable ASCII characters (0–31) or the following special characters: *<>\=
value	String	Specifies the key value. The value can contain a maximum of 43 Unicode characters and can be an empty string. It cannot contain ASCII characters (0–31) and the following characters: *<>\=

Examples

- Example request

  a. Request body when **action** is set to **filter**

```
POST  https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/
resource_instances/action
{
   "offset": "100",
   "limit": "100",
   "action": "filter",
   "matches": [
      {
         "key": "resource_name",
         "value": "nat_gateways"
      }
   ],
   "not_tags": [
      {
         "key": "key1",
         "values": [
            "*value1",
            "value2"
         ]
      }
   ],
   "tags": [
      {
         "key": "key2",
         "values": [
            "*value3",
            "value4"
         ]
      }
   ],
   "tags_any": [
      {
         "key": "key3",
         "values": [
```



```
            "*value5",
            "value6"
         ]
      }
   ],
   "not_tags_any": [
      {
         "key": "key4",
         "values": [
            "*value7",
            "value8"
         ]
      }
   ]
}
```

b. Request body when **action** is set to **count**

```
POST  https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/
resource_instances/action        {
   "action": "count",
   "matches": [
      {
         "key": "resource_name",
         "value": "nat_gateways"
      }
   ],
   "not_tags": [
      {
         "key": "key1",
         "values": [
            "*value1",
            "value2"
         ]
      }
   ],
   "tags": [
      {
         "key": "key2",
         "values": [
            "*value3",
            "value4"
         ]
      }
   ],
   "tags_any": [
      {
         "key": "key3",
         "values": [
            "*value5",
            "value6"
         ]
      }
   ],
   "not_tags_any": [
      {
         "key": "key4",
         "values": [
            "*value7",
            "value8"
         ]
      }
   ]
}
```

- Example response

a. Response body when **action** is set to **filter**

```
{
   "resources": [
```



```
    {
      "resource_detail": null,
      "resource_id": "e5ad289f-9c56-4daf-b08b-2e53a983473a",
      "resource_name": "nat_gateways",
      "tags": [
        {
          "key": "key2",
          "value": "value4"
        },
        {
          "key": "key2",
          "value": "value3"
        }
      ]
    }
  ],
  "total_count": 1000
}
```


b. Response body when **action** is set to **count**


```
{
  "total_count": 1000
}
```


Status Codes

See **Status Codes**.

5.1.2 Adding or Deleting NAT Gateway Tags in Batches

Function

This API is used to add or delete tags of a specific NAT gateway in batches.

TMS uses this API to manage resource tags.

You can add a maximum of 10 tags to a NAT gateway.

This API is idempotent.

If there are duplicate keys in the request body when you add tags, an error is reported.

If a to-be-created tag has the same key as an existing tag, the tag will be created and overwrite the existing one.

When tags are being deleted and some tags do not exist, the operation is considered to be successful by default. The character set of the tags will not be checked. A key and a value can respectively contain up to 127 and 255 Unicode characters. When you delete tags, the tag structure cannot be missing, and the key cannot be left blank or be an empty string.

URI

- URI format

POST /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags/action

Table 5-8 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.

Request

Table 5-9 describes the request parameters.

Table 5-9 Request parameter

Parameter	Mandatory	Type	Description
tags	Yes	Array	Specifies tags. For details, see Table 5-10.
action	Yes	String	Specifies the operation to be performed, which can be set to **create** or **delete** only.

Table 5-10 Parameter description of field **tags**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. It contains a maximum of 36 Unicode characters. A tag key cannot be left blank. It cannot contain ASCII (0-31) or the following characters: =*<>\,|/
value	Yes	String	Specifies the key value. Each value contains a maximum of 43 Unicode characters. If **value** is specified, tags are deleted by key and value. If **value** is not specified, tags are deleted by key. The value can be an empty character string. It cannot contain ASCII (0-31) or the following characters: =*<>\,|/

Response

None

Examples

- Example request

```
POST  https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/
fe1a4cf0-27fe-4b97-a9b1-2c67c127f0e0/tags/action
{
   "action": "create",
   "tags": [
      {
         "key": "key1",
         "value": "value1"
      },
      {
         "key": "key2",
         "value": "value2"
      }
   ]
}
Or
{
   "action": "delete",
   "tags": [
      {
         "key": "key1",
         "value": "value1
       },
      {
         "key": "key2",
         "value": "value2"
      }
   ]
}
```

Example response

None

Status Codes

See **Status Codes**.

5.1.3 Adding a NAT Gateway Tag

Function

This API is used to add tags to a NAT gateway. You can add a maximum of 10 tags to a NAT gateway.

This API is idempotent.

If a to-be-created tag has the same key as an existing tag, the tag will be created and overwrite the existing one.

NOTE

Ensure that the NAT gateway to which you will add tags is available.

URI

POST /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags

Table 5-11 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.

Request

Table 5-12 describes the request parameters.

Table 5-12 Request parameter

Parameter	Mandatory	Type	Description
tag	Yes	Array	Specifies the list of tags.

Table 5-13 Parameter description of field **tag**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. It contains a maximum of 36 Unicode characters. A tag key cannot be left blank. It cannot contain ASCII (0-31) or the following characters: =*<>\,|/
value	Yes	String	Specifies the key value. The value can contain a maximum of 43 Unicode characters and can be an empty string. It cannot contain ASCII (0-31) or the following characters: =*<>\,|/

Response

None

Examples

- Example request

```
POST https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/fe1a4cf0-27fe-4b97-a9b1-2c67c127f0e0/tags
{
    "tag":
   {
```



```
        "key":"key1",
        "value":"value1"
    }
}
```

Status Codes

See **Status Codes**.

5.1.4 Deleting a NAT Gateway Tag

Function

This API is idempotent.

When a tag is deleted, it is not verified. The tag key cannot be left blank or be an empty string. If the key of the tag to be deleted is not found, 404 will be returned.

URI

DELETE /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags/{key}

Table 5-14 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.
key	Yes	String	Specifies the tag key.

Request

None

Response

None

Examples

- Example request
  ```
  DELETE https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/fe1a4cf0-27fe-4b97-a9b1-2c67c127f0e0/tags/key1
  ```
- Example response
  ```
  None (STATUS CODE 204)
  ```

Status Codes

See **Status Codes**.

5.1.5 Querying NAT Gateway Tags

Function

This API is used to query tags of a specified NAT gateway.

TMS uses this API to query all tags of a specified NAT gateway.

URI

GET /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags

Table 5-15 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.
nat_gateway_id	Yes	String	Specifies the NAT gateway ID.

Request

None

Response

Table 5-16 describes the response parameters.

Table 5-16 Response parameter

Parameter	Mandatory	Type	Description
tags	Yes	Array	Specifies the list of tags.

Table 5-17 Parameter description of field **tags**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. It contains a maximum of 36 Unicode characters. A tag key cannot be left blank. It cannot contain non-printable ASCII characters (0–31) or the following special characters: *<>\=

Parameter	Mandatory	Type	Description
value	Yes	String	Specifies the key value. The value can contain a maximum of 43 Unicode characters and can be an empty string. It cannot contain non-printable ASCII characters (0–31) or the following special characters: *<>\=

Examples

- Example request
  ```
  GET https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/fe1a4cf0-27fe-4b97-a9b1-2c67c127f0e0/tags
  ```
- Example response
  ```
  {
     "tags": [
        {
           "key": "key1",
           "value": "value1"
        },
        {
           "key": "key2",
           "value": "value2"
        }
     ]
  }
  ```

Status Codes

See **Status Codes**.

5.1.6 Querying Tags in a Project

Function

This API is used to query all tags of a resource type in a specified region.

TMS uses this API to list tags created by a tenant to facilitate tag creation and resource filtering on the console.

URI

GET /v2.0/{project_id}/nat_gateways/tags

Table 5-18 Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID.

Request

None

Response

Table 5-19 describes the response parameters.

Table 5-19 Response parameter

Parameter	Mandatory	Type	Description
tags	Yes	Array	Specifies the list of tags.

Table 5-20 Parameter description of field **tags**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. It contains a maximum of 36 Unicode characters. A tag key cannot be left blank. It cannot contain non-printable ASCII characters (0–31) or the following special characters: *<>\=
values	Yes	Array<String>	Specifies the tag values. The value can contain a maximum of 43 Unicode characters and can be an empty string. It cannot contain non-printable ASCII characters (0–31) or the following special characters: *<>\=

Examples

- Example request
  ```
  GET https://{VPC_endpoint}/v2.0/9ad601814ac94c80bf7bb9073ded66fc/nat_gateways/tags
  ```
- Example response
  ```
  {
    "tags": [
      {
        "key": "key1",
        "values": [
          "value1",
          "value2"
        ]
      },
  ```

```
    {
      "key": "key2",
      "values": [
        "value3",
        "value4"
      ]
    }
  ]
}
```

Status Codes

See **Status Codes**.

6 Private Nat API

6.1 Private NAT Gateways

6.1.1 Querying Private NAT Gateways

Function

This API is used to query private NAT gateways.

Constraints

You can type the question mark (?) and ampersand (&) at the end of the URI to define multiple search criteria. All optional parameters can be filtered. For details, see the example request.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/gateways

Table 6-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Table 6-2 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the number of records displayed on each page. The value ranges from 0 to 2000. Default value: 2000 Minimum: **1** Maximum: **2000** Default: **2000**
marker	No	String	Specifies the start resource ID of pagination query. If the parameter is left blank, only resources on the first page are queried. The value is obtained from next_marker or previous_marker in PageInfo queried last time. Minimum: **36** Maximum: **36**
page_reverse	No	Boolean	Specifies whether to query resources on the previous page.
id	No	Array	Specifies the private NAT gateway ID.
name	No	Array	Specifies the private NAT gateway name.
description	No	Array	Provides supplementary information about the private NAT gateway.
spec	No	Array	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**
project_id	No	Array	Specifies the project ID.

Parameter	Mandatory	Type	Description
status	No	Array	Specifies the private NAT gateway status. The value can be: ACTIVE: The private NAT gateway is running properly. FROZEN: The private NAT gateway is frozen. Enumeration values: • **ACTIVE** • **FROZEN**
vpc_id	No	Array	Specifies the ID of the VPC where the private NAT gateway resides.
virsubnet_id	No	Array	Specifies the ID of the subnet where the private NAT gateway resides.
enterprise_project_id	No	Array	Specifies the ID of the enterprise project that is associated with the private NAT gateway when the private NAT gateway is created.

Request Parameters

Table 6-3 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-4 Response body parameters

Parameter	Type	Description
gateways	Array of **PrivateNat** objects	Specifies the response body for querying private NAT gateways. For details, see the PrivateNat description.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
page_info	**PageInfo** object	Specifies the pagination information.

Table 6-5 PrivateNat

Parameter	Type	Description
id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **32** Maximum: **32**
name	String	Specifies the private NAT gateway name. Minimum: **1** Maximum: **64**
description	String	Provides supplementary information about the private NAT gateway. Minimum: **0** Maximum: **255**
spec	String	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Default: **Small** Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**

Parameter	Type	Description
status	String	Specifies the private NAT gateway status. The value can be: ACTIVE: The private NAT gateway is running properly. FROZEN: The private NAT gateway is frozen. Enumeration values: • **ACTIVE** • **FROZEN**
created_at	String	Specifies when the private NAT gateway was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
updated_at	String	Specifies when the private NAT gateway was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
downlink_vpcs	Array of **DownlinkVpc** objects	Specifies the VPC where the private NAT gateway resides.
tags	Array of **Tag** objects	Specifies the list of tags.
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the private NAT gateway when the private NAT gateway is created. Minimum: **1** Maximum: **36**

Table 6-6 DownlinkVpc

Parameter	Type	Description
vpc_id	String	Specifies the ID of the VPC where the private NAT gateway resides. Minimum: **36** Maximum: **36**
virsubnet_id	String	Specifies the ID of the subnet where the private NAT gateway resides. Minimum: **36** Maximum: **36**

Table 6-7 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Table 6-8 PageInfo

Parameter	Type	Description
next_marker	String	Specifies the ID of the last record in this query, which can be used in the next query. Minimum: **1** Maximum: **36**
previous_marker	String	Specifies the ID of the first record in the pagination query result. When page_reverse is set to true, this parameter is used together to query resources on the previous page. Minimum: **1** Maximum: **36**
current_count	Integer	Specifies the ID of the last record in the pagination query result. It is usually used to query resources on the next page. Minimum: **1** Maximum: **2000**

Example Requests

None

Example Responses

Status code: 200

Private NAT gateways queried.

```
{
  "gateways" : [ {
    "id" : "14338426-6afe-4019-996b-3a9525296e11",
    "name" : "private-nat-gateway-name1",
    "description" : "private-nat-gateway-description1",
    "spec" : "Small",
```



```
  "project_id" : "70505c941b9b4dfd82fd351932328a2f",
  "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
  "status" : "ACTIVE",
  "created_at" : "2019-04-22T08:47:13",
  "updated_at" : "2019-04-22T08:47:13",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  } ],
  "downlink_vpcs" : [ {
    "vpc_id" : "3cb66d44-9f75-4237-bfff-e37b14d23ad2",
    "virsubnet_id" : "373979ee-f4f0-46c5-80e3-0fbf72646b70"
  } ]
 }, {
  "id" : "65995b8e-dcb7-4ab4-9931-bc3c95beec0a",
  "name" : "private-nat-gateway-name2",
  "description" : "private-nat-gateway-description2",
  "spec" : "Small",
  "project_id" : "70505c941b9b4dfd82fd351932328a2f",
  "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
  "status" : "ACTIVE",
  "created_at" : "2019-04-22T09:06:54",
  "updated_at" : "2019-04-22T09:06:54",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  } ],
  "downlink_vpcs" : [ {
    "vpc_id" : "3cb66d44-9f75-4237-bfff-e37b14d23ad2",
    "virsubnet_id" : "373979ee-f4f0-46c5-80e3-0fbf72646b70"
  } ]
 } ],
 "request_id" : "a7b00469-5a31-4274-bb10-59167243383e",
 "page_info" : {
  "previous_marker" : "14338426-6afe-4019-996b-3a9525296e11",
  "current_count" : 2
 }
}
```


Status Codes

Status Code	Description
200	Private NAT gateways queried.

Error Codes

See **Error Codes**.

6.1.2 Updating a Private NAT Gateway

Function

This API is used to update a private NAT gateway.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

PUT /v3/{project_id}/private-nat/gateways/{gateway_id}

Table 6-9 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
gateway_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-10 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Table 6-11 Request body parameters

Parameter	Mandatory	Type	Description
gateway	Yes	**UpdatePrivateNatOption** object	Specifies the request body for updating the private NAT gateway.

Table 6-12 UpdatePrivateNatOption

Parameter	Mandatory	Type	Description
name	No	String	Specifies the private NAT gateway name. Only digits, letters, underscores (_), and hyphens (-) are allowed. Minimum: **1** Maximum: **64**
description	No	String	Provides supplementary information about the private NAT gateway. Minimum: **0** Maximum: **255**
spec	No	String	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**

Response Parameters

Status code: 200

Table 6-13 Response body parameters

Parameter	Type	Description
gateway	**PrivateNat** object	Specifies the response body for the private NAT gateway.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-14 PrivateNat

Parameter	Type	Description
id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **32** Maximum: **32**
name	String	Specifies the private NAT gateway name. Minimum: **1** Maximum: **64**
description	String	Provides supplementary information about the private NAT gateway. Minimum: **0** Maximum: **255**
spec	String	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Default: **Small** Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**
status	String	Specifies the private NAT gateway status. The value can be: ACTIVE: The private NAT gateway is running properly. FROZEN: The private NAT gateway is frozen. Enumeration values: • **ACTIVE** • **FROZEN**
created_at	String	Specifies when the private NAT gateway was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
updated_at	String	Specifies when the private NAT gateway was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
downlink_vpcs	Array of **DownlinkVpc** objects	Specifies the VPC where the private NAT gateway resides.

Parameter	Type	Description
tags	Array of **Tag** objects	Specifies the list of tags.
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the private NAT gateway when the private NAT gateway is created. Minimum: **1** Maximum: **36**

Table 6-15 DownlinkVpc

Parameter	Type	Description
vpc_id	String	Specifies the ID of the VPC where the private NAT gateway resides. Minimum: **36** Maximum: **36**
virsubnet_id	String	Specifies the ID of the subnet where the private NAT gateway resides. Minimum: **36** Maximum: **36**

Table 6-16 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

```
PUT https://{Endpoint}/v3/70505c941b9b4dfd82fd351932328a2f/private-nat/gateways/
14338426-6afe-4019-996b-3a9525296e11

{
  "gateway" : {
    "name" : "private-nat-gateway-name",
    "description" : "private-nat-gateway-description",
    "spec" : "Medium"
  }
}
```


Example Responses

Status code: 200

Private NAT gateway updated.

```
{
  "gateway" : {
    "id" : "14338426-6afe-4019-996b-3a9525296e11",
    "name" : "private-nat-gateway-name",
    "description" : "private-nat-gateway-description",
    "spec" : "Medium",
    "project_id" : "70505c941b9b4dfd82fd351932328a2f",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "status" : "ACTIVE",
    "created_at" : "2019-04-22T08:47:13",
    "updated_at" : "2019-04-22T08:47:13",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ],
    "downlink_vpcs" : [ {
      "vpc_id" : "3cb66d44-9f75-4237-bfff-e37b14d23ad2",
      "virsubnet_id" : "373979ee-f4f0-46c5-80e3-0fbf72646b70"
    } ]
  },
  "request_id" : "e7e3323e95b348708d26e68a0ddece71"
}
```

Status Codes

Status Code	Description
200	Private NAT gateway updated.

Error Codes

See **Error Codes**.

6.1.3 Deleting a Private NAT Gateway

Function

This API is used to delete a private NAT gateway.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

DELETE /v3/{project_id}/private-nat/gateways/{gateway_id}

Table 6-17 Path Parameters

Parameter	Mandatory	Type	Description
gateway_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Request Parameters

Table 6-18 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

None

Example Requests

None

Example Responses

None

Status Codes

Status Code	Description
204	Private NAT gateway deleted.

Error Codes

See **Error Codes**.

6.1.4 Creating a Private NAT Gateway

Function

This API is used to create a private NAT gateway.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat/gateways

Table 6-19 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Request Parameters

Table 6-20 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Table 6-21 Request body parameters

Parameter	Mandatory	Type	Description
gateway	Yes	**CreatePrivateNatOption** object	Specifies the request body for creating the private NAT gateway.

Table 6-22 CreatePrivateNatOption

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the private NAT gateway name. Only digits, letters, underscores (_), and hyphens (-) are allowed. Minimum: **1** Maximum: **64**
description	No	String	Provides supplementary information about the private NAT gateway. Minimum: **0** Maximum: **255**

Parameter	Mandatory	Type	Description
spec	No	String	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Default: **Small** Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**
downlink_vpcs	Yes	Array of **DownlinkVpcOption** objects	Specifies the VPC where the private NAT gateway resides.
tags	No	Array of **Tag** objects	Specifies the tag list.
enterprise_project_id	No	String	Specifies the ID of the enterprise project that is associated with the private NAT gateway when the private NAT gateway is created. For more information about enterprise projects and how to obtain enterprise project IDs, see Enterprise Management User Guide. Default: **0** Minimum: **1** Maximum: **36**

Table 6-23 DownlinkVpcOption

Parameter	Mandatory	Type	Description
virsubnet_id	Yes	String	Specifies the ID of the subnet where the private NAT gateway resides. Minimum: **36** Maximum: **36**

Table 6-24 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Response Parameters

Status code: 201

Table 6-25 Response body parameters

Parameter	Type	Description
gateway	**PrivateNat** object	Specifies the response body for the private NAT gateway.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-26 PrivateNat

Parameter	Type	Description
id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **32** Maximum: **32**
name	String	Specifies the private NAT gateway name. Minimum: **1** Maximum: **64**
description	String	Provides supplementary information about the private NAT gateway. Minimum: **0** Maximum: **255**

Parameter	Type	Description
spec	String	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Default: **Small** Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**
status	String	Specifies the private NAT gateway status. The value can be: ACTIVE: The private NAT gateway is running properly. FROZEN: The private NAT gateway is frozen. Enumeration values: • **ACTIVE** • **FROZEN**
created_at	String	Specifies when the private NAT gateway was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
updated_at	String	Specifies when the private NAT gateway was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
downlink_vpcs	Array of **DownlinkVpc** objects	Specifies the VPC where the private NAT gateway resides.
tags	Array of **Tag** objects	Specifies the list of tags.
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the private NAT gateway when the private NAT gateway is created. Minimum: **1** Maximum: **36**

Table 6-27 DownlinkVpc

Parameter	Type	Description
vpc_id	String	Specifies the ID of the VPC where the private NAT gateway resides. Minimum: **36** Maximum: **36**

Parameter	Type	Description
virsubnet_id	String	Specifies the ID of the subnet where the private NAT gateway resides. Minimum: **36** Maximum: **36**

Table 6-28 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

This API is used to create a private NAT gateway.

```
POST https://{Endpoint}/v3/70505c941b9b4dfd82fd351932328a2f/private-nat/gateways

{
  "gateway" : {
    "name" : "private-nat-gateway-name",
    "description" : "private-nat-gateway-description",
    "spec" : "Small",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "downlink_vpcs" : [ {
      "virsubnet_id" : "373979ee-f4f0-46c5-80e3-0fbf72646b70"
    } ],
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ]
  }
}
```

Example Responses

Status code: 201

Private NAT gateway created.

```
{
  "request_id" : "9882046a9b96f1405472e36d797e33dc",
  "gateway" : {
    "id" : "14338426-6afe-4019-996b-3a9525296e11",
    "name" : "private-nat-gateway-name",
    "description" : "private-nat-gateway-description",
    "spec" : "Small",
    "project_id" : "70505c941b9b4dfd82fd351932328a2f",
```



```
  "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
  "status" : "ACTIVE",
  "created_at" : "2019-04-22T08:47:13",
  "updated_at" : "2019-04-22T08:47:13",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  } ],
  "downlink_vpcs" : [ {
    "vpc_id" : "3cb66d44-9f75-4237-bfff-e37b14d23ad2",
    "virsubnet_id" : "373979ee-f4f0-46c5-80e3-0fbf72646b70"
  } ]
 }
}
```


Status Codes

Status Code	Description
201	Private NAT gateway created.

Error Codes

See **Error Codes**.

6.1.5 Querying Details About a Specified Private NAT Gateway

Function

This API is used to query details about a specified private NAT gateway.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/gateways/{gateway_id}

Table 6-29 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
gateway_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-30 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-31 Response body parameters

Parameter	Type	Description
gateway	**PrivateNat** object	Specifies the response body for the private NAT gateway.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-32 PrivateNat

Parameter	Type	Description
id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **32** Maximum: **32**

Parameter	Type	Description
name	String	Specifies the private NAT gateway name. Minimum: **1** Maximum: **64**
description	String	Provides supplementary information about the private NAT gateway. Minimum: **0** Maximum: **255**
spec	String	Specifies the private NAT gateway type. The value can be: Small Medium Large Extra-large Default: **Small** Enumeration values: • **Small** • **Medium** • **Large** • **Extra-large**
status	String	Specifies the private NAT gateway status. The value can be: ACTIVE: The private NAT gateway is running properly. FROZEN: The private NAT gateway is frozen. Enumeration values: • **ACTIVE** • **FROZEN**
created_at	String	Specifies when the private NAT gateway was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
updated_at	String	Specifies when the private NAT gateway was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format.
downlink_vpcs	Array of **DownlinkVpc** objects	Specifies the VPC where the private NAT gateway resides.
tags	Array of **Tag** objects	Specifies the list of tags.
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the private NAT gateway when the private NAT gateway is created. Minimum: **1** Maximum: **36**

Table 6-33 DownlinkVpc

Parameter	Type	Description
vpc_id	String	Specifies the ID of the VPC where the private NAT gateway resides. Minimum: **36** Maximum: **36**
virsubnet_id	String	Specifies the ID of the subnet where the private NAT gateway resides. Minimum: **36** Maximum: **36**

Table 6-34 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

None

Example Responses

Status code: 200

Details about the private NAT gateway queried.


```
{
  "gateway" : {
    "id" : "14338426-6afe-4019-996b-3a9525296e11",
    "name" : "private-nat-gateway-name",
    "description" : "private-nat-gateway-description",
    "spec" : "Small",
    "project_id" : "70505c941b9b4dfd82fd351932328a2f",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "status" : "ACTIVE",
    "created_at" : "2019-04-22T08:47:13",
    "updated_at" : "2019-04-22T08:47:13",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ],
    "downlink_vpcs" : [ {
      "vpc_id" : "3cb66d44-9f75-4237-bfff-e37b14d23ad2",
```



```
    "virsubnet_id" : "373979ee-f4f0-46c5-80e3-0fbf72646b70"
  } ]
 },
 "request_id" : "747a911c17067a39692f75ac146fb47e"
}
```

Status Codes

Status Code	Description
200	Details about the private NAT gateway queried.

Error Codes

See **Error Codes**.

6.2 DNAT Rules

6.2.1 Querying DNAT Rules

Function

This API is used to query DNAT rules.

Constraints

You can type the question mark (?) and ampersand (&) at the end of the URI to define multiple search criteria. All optional parameters can be filtered. For details, see the example request.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/dnat-rules

Table 6-35 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Table 6-36 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the number of records displayed on each page. The value ranges from 0 to 2000. Default value: 2000 Minimum: **1** Maximum: **2000** Default: **2000**
marker	No	String	Specifies the start resource ID of pagination query. If the parameter is left blank, only resources on the first page are queried. The value is obtained from next_marker or previous_marker in PageInfo queried last time. Minimum: **36** Maximum: **36**
page_reverse	No	Boolean	Specifies whether to query the previous page.
id	No	Array	Specifies the DNAT rule ID.
project_id	No	Array	Specifies the project ID.
enterprise_project_id	No	Array	Specifies the ID of the enterprise project that is associated with the DNAT rule when the DNAT rule is being created.
description	No	Array	Provides supplementary information about the DNAT rule.
gateway_id	No	Array	Specifies the private NAT gateway ID.
transit_ip_id	No	Array	Specifies the ID of the transit IP address.
external_ip_address	No	Array	Specifies the transit IP address.
network_interface_id	No	Array	Specifies the network interface ID. Network interfaces of an Elastic Cloud Server (ECS), load balancer, or virtual IP address are supported.

Parameter	Mandatory	Type	Description
type	No	Array	Specifies the backend resource type of the DNAT rule. The type can be: COMPUTE: The backend resource is a compute instance. VIP: The backend resource is a virtual IP address. ELB: The backend resource is a shared load balancer. ELBv3: The backend resource is a dedicated load balancer. CUSTOMIZE: The backend resource is a user-defined IP address.
private_ip_address	No	Array	Specifies the private IP address of the backend instance.

Request Parameters

Table 6-37 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-38 Response body parameters

Parameter	Type	Description
dnat_rules	Array of **PrivateDnat** objects	Specifies the response body for querying DNAT rules.

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
page_info	**PageInfo** object	Specifies the pagination information.

Table 6-39 PrivateDnat

Parameter	Type	Description
id	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the DNAT rule. Minimum: **1** Maximum: **36**
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **1** Maximum: **36**
network_interface_id	String	Specifies the network interface ID. Network interfaces of a compute instance, load balancer, virtual IP address are supported. Minimum: **1** Maximum: **36**

Parameter	Type	Description
type	String	Specifies the backend resource type of the DNAT rule. The type can be: COMPUTE: The backend resource is a compute instance. VIP: The backend resource is a virtual IP address. ELB: The backend resource is a load balancer. ELBv3: The backend resource is a dedicated load balancer. CUSTOMIZE: The backend resource is a user-defined IP address. Minimum: **1** Maximum: **10**
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively. Minimum: **1** Maximum: **3** Enumeration values: • **tcp** • **udp** • **any**
private_ip_address	String	Specifies the private IP address of the backend instance. Minimum: **7** Maximum: **15**
internal_service_port	String	Specifies the port number of the backend instance. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
transit_service_port	String	Specifies the port number of the transit IP address. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the DNAT rule when the DNAT rule is being created. Minimum: **1** Maximum: **36**

Parameter	Type	Description
created_at	String	Specifies when the DNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the DNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Table 6-40 PageInfo

Parameter	Type	Description
next_marker	String	Specifies the ID of the last record in this query, which can be used in the next query. Minimum: **1** Maximum: **36**
previous_marker	String	Specifies the ID of the first record in the pagination query result. When page_reverse is set to true, this parameter is used together to query resources on the previous page. Minimum: **1** Maximum: **36**
current_count	Integer	Specifies the ID of the last record in the pagination query result. It is usually used to query resources on the next page. Minimum: **1** Maximum: **2000**

Example Requests

None

Example Responses

Status code: 200

DNAT rules queried.

```
{
  "dnat_rules" : [ {
    "id" : "24dd6bf5-48f2-4915-ad0b-5bb111d39c83",
```



```
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "description" : "aa",
    "gateway_id" : "0adefb29-a6c2-48a5-8637-2be67fa03fec",
    "transit_ip_id" : "3faa719d-6d18-4ccb-a5c7-33e65a09663e",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "network_interface_id" : "dae9393a-b536-491c-a5a2-72edc1104707",
    "type" : "COMPUTE",
    "protocol" : "any",
    "internal_service_port" : "0",
    "transit_service_port" : "0",
    "private_ip_address" : "192.168.1.72",
    "created_at" : "2019-04-29T07:10:01",
    "updated_at" : "2019-04-29T07:10:01"
  }, {
    "id" : "25dcdb21-97de-43cd-b476-31637a47f05d",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "description" : "aa",
    "gateway_id" : "0adefb29-a6c2-48a5-8637-2be67fa03fec",
    "transit_ip_id" : "15abdf29-4a68-474c-9963-79c4e6d495d7",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "network_interface_id" : "9e2f0dbb-68b2-4c4b-9298-fa4f13187976",
    "type" : "COMPUTE",
    "protocol" : "any",
    "internal_service_port" : "0",
    "transit_service_port" : "0",
    "private_ip_address" : "192.168.1.99",
    "created_at" : "2019-04-29T07:15:41",
    "updated_at" : "2019-04-29T07:15:41"
  } ],
  "request_id" : "a7b00469-5a31-4274-bb10-59167243383e",
  "page_info" : {
    "previous_marker" : "14338426-6afe-4019-996b-018008113013",
    "current_count" : 2
  }
}
```


Status Codes

Status Code	Description
200	DNAT rules queried.

Error Codes

See **Error Codes**.

6.2.2 Updating a DNAT Rule

Function

This API is used to update a specified DNAT rule.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

PUT /v3/{project_id}/private-nat/dnat-rules/{dnat_rule_id}

Table 6-41 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
dnat_rule_id	Yes	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-42 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Table 6-43 Request body parameters

Parameter	Mandatory	Type	Description
dnat_rule	No	**UpdatePrivateDnatOption** object	Specifies the request body for updating a DNAT rule.

Table 6-44 UpdatePrivateDnatOption

Parameter	Mandatory	Type	Description
description	No	String	Provides supplementary information about the DNAT rule. Minimum: **0** Maximum: **255**
transit_ip_id	No	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
network_interface_id	No	String	Specifies the network interface ID. Network interfaces of an ECS, load balancer, or virtual IP address are supported. Minimum: **36** Maximum: **36**
private_ip_address	No	String	Specifies the private IP address of the backend instance. Minimum: **7** Maximum: **15**
protocol	No	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively. Minimum: **1** Maximum: **3** Enumeration values: • **tcp** • **udp** • **any**
internal_service_port	No	String	Specifies the port number of the backend instance. Minimum: **1** Maximum: **5**
transit_service_port	No	String	Specifies the port number of the transit IP address. Minimum: **1** Maximum: **10**

Response Parameters

Status code: 200

Table 6-45 Response body parameters

Parameter	Type	Description
dnat_rule	**PrivateDnat** object	Specifies the response body of the DNAT rule.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-46 PrivateDnat

Parameter	Type	Description
id	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the DNAT rule. Minimum: **1** Maximum: **36**
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **1** Maximum: **36**
network_interface_id	String	Specifies the network interface ID. Network interfaces of a compute instance, load balancer, virtual IP address are supported. Minimum: **1** Maximum: **36**

Parameter	Type	Description
type	String	Specifies the backend resource type of the DNAT rule. The type can be: COMPUTE: The backend resource is a compute instance. VIP: The backend resource is a virtual IP address. ELB: The backend resource is a load balancer. ELBv3: The backend resource is a dedicated load balancer. CUSTOMIZE: The backend resource is a user-defined IP address. Minimum: **1** Maximum: **10**
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively. Minimum: **1** Maximum: **3** Enumeration values: • **tcp** • **udp** • **any**
private_ip_address	String	Specifies the private IP address of the backend instance. Minimum: **7** Maximum: **15**
internal_service_port	String	Specifies the port number of the backend instance. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
transit_service_port	String	Specifies the port number of the transit IP address. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the DNAT rule when the DNAT rule is being created. Minimum: **1** Maximum: **36**

Parameter	Type	Description
created_at	String	Specifies when the DNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the DNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Example Requests

```
PUT https://{Endpoint}/v3/da261828016849188f4dcc2ef94d9da9/private-nat/dnat-rules/
24dd6bf5-48f2-4915-ad0b-5bb111d39c83

{
  "dnat_rule" : {
    "description" : "my dnat-rules 03"
  }
}
```

Example Responses

Status code: 200

DNAT rule updated.

```
{
  "dnat_rule" : {
    "id" : "24dd6bf5-48f2-4915-ad0b-5bb111d39c83",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "description" : "dnat rule description",
    "gateway_id" : "0adefb29-a6c2-48a5-8637-2be67fa03fec",
    "transit_ip_id" : "3faa719d-6d18-4ccb-a5c7-33e65a09663e",
    "network_interface_id" : "dae9393a-b536-491c-a5a2-72edc1104707",
    "type" : "COMPUTE",
    "private_ip_address" : "192.168.1.72",
    "created_at" : "2019-04-29T07:10:01",
    "updated_at" : "2019-04-29T07:10:01"
  },
  "request_id" : "747a911c17067a39692f75ac146fb47e"
}
```

Status Codes

Status Code	Description
200	DNAT rule updated.

Error Codes

See **Error Codes**.

6.2.3 Creating a DNAT Rule

Function

This API is used to create a DNAT rule.

Constraints

When you are creating a DNAT rule, status of the NAT gateway must be set to ACTIVE.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat/dnat-rules

Table 6-47 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Request Parameters

Table 6-48 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Table 6-49 Request body parameters

Parameter	Mandatory	Type	Description
dnat_rule	Yes	CreatePrivateDnatOption object	Specifies the request body for creating a DNAT rule.

Table 6-50 CreatePrivateDnatOption

Parameter	Mandatory	Type	Description
description	No	String	Provides supplementary information about the DNAT rule. Minimum: **0** Maximum: **255**
transit_ip_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
network_interface_id	No	String	Specifies the network interface ID. Network interfaces of an ECS, load balancer, or virtual IP address are supported. Minimum: **36** Maximum: **36**
gateway_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
protocol	No	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively. Minimum: **1** Maximum: **3** Enumeration values: • **tcp** • **udp** • **any**

Parameter	Mandatory	Type	Description
private_ip_address	No	String	Specifies the private IP address of the backend instance. Minimum: **7** Maximum: **15**
internal_service_port	No	String	Specifies the port number of the backend instance. Minimum: **1** Maximum: **5**
transit_service_port	No	String	Specifies the port number of the transit IP address. Minimum: **1** Maximum: **5**

Response Parameters

Status code: 201

Table 6-51 Response body parameters

Parameter	Type	Description
dnat_rule	**PrivateDnat** object	Specifies the response body of the DNAT rule.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-52 PrivateDnat

Parameter	Type	Description
id	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**

Parameter	Type	Description
description	String	Provides supplementary information about the DNAT rule. Minimum: **1** Maximum: **36**
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **1** Maximum: **36**
network_interface_id	String	Specifies the network interface ID. Network interfaces of a compute instance, load balancer, virtual IP address are supported. Minimum: **1** Maximum: **36**
type	String	Specifies the backend resource type of the DNAT rule. The type can be: COMPUTE: The backend resource is a compute instance. VIP: The backend resource is a virtual IP address. ELB: The backend resource is a load balancer. ELBv3: The backend resource is a dedicated load balancer. CUSTOMIZE: The backend resource is a user-defined IP address. Minimum: **1** Maximum: **10**
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively. Minimum: **1** Maximum: **3** Enumeration values: • **tcp** • **udp** • **any**
private_ip_address	String	Specifies the private IP address of the backend instance. Minimum: **7** Maximum: **15**

Parameter	Type	Description
internal_service_port	String	Specifies the port number of the backend instance. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
transit_service_port	String	Specifies the port number of the transit IP address. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the DNAT rule when the DNAT rule is being created. Minimum: **1** Maximum: **36**
created_at	String	Specifies when the DNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the DNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Example Requests

```
POST https://{Endpoint}/v3/da261828016849188f4dcc2ef94d9da9/private-nat/dnat-rules

{
  "dnat_rule" : {
    "description" : "aa",
    "gateway_id" : "0adefb29-a6c2-48a5-8637-2be67fa03fec",
    "transit_ip_id" : "3faa719d-6d18-4ccb-a5c7-33e65a09663e",
    "network_interface_id" : "dae9393a-b536-491c-a5a2-72edc1104707"
  }
}
```

Example Responses

Status code: 201

DNAT rule created.


```
{
  "dnat_rule" : {
    "id" : "24dd6bf5-48f2-4915-ad0b-5bb111d39c83",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "description" : "aa",
    "gateway_id" : "0adefb29-a6c2-48a5-8637-2be67fa03fec",
    "transit_ip_id" : "3faa719d-6d18-4ccb-a5c7-33e65a09663e",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "network_interface_id" : "dae9393a-b536-491c-a5a2-72edc1104707",
    "type" : "COMPUTE",
    "protocol" : "any",
    "internal_service_port" : "0",
    "transit_service_port" : "0",
    "private_ip_address" : "192.168.1.72",
    "created_at" : "2019-04-29T07:10:01",
    "updated_at" : "2019-04-29T07:10:01"
  },
  "request_id" : "70505c941b9b4dfd82fd351932328a2f"
}
```


Status Codes

Status Code	Description
201	DNAT rule created.

Error Codes

See **Error Codes**.

6.2.4 Deleting a DNAT Rule

Function

This API is used to delete a specified DNAT rule.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

DELETE /v3/{project_id}/private-nat/dnat-rules/{dnat_rule_id}

Table 6-53 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Parameter	Mandatory	Type	Description
dnat_rule_id	Yes	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-54 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

None

Example Requests

```
DELETE https://{Endpoint}/v3/da261828016849188f4dcc2ef94d9da9/private-nat/dnat-rules/24dd6bf5-48f2-4915-ad0b-5bb111d39c83
```

Example Responses

None

Status Codes

Status Code	Description
204	DNAT rule deleted.

Error Codes

See **Error Codes**.

6.2.5 Querying Details About a Specified DNAT Rule

Function

This API is used to query details about a specified DNAT rule.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/dnat-rules/{dnat_rule_id}

Table 6-55 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
dnat_rule_id	Yes	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-56 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-57 Response body parameters

Parameter	Type	Description
dnat_rule	**PrivateDnat** object	Specifies the response body of the DNAT rule.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-58 PrivateDnat

Parameter	Type	Description
id	String	Specifies the DNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the DNAT rule. Minimum: **1** Maximum: **36**
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **1** Maximum: **36**
network_interface_id	String	Specifies the network interface ID. Network interfaces of a compute instance, load balancer, virtual IP address are supported. Minimum: **1** Maximum: **36**

Parameter	Type	Description
type	String	Specifies the backend resource type of the DNAT rule. The type can be: COMPUTE: The backend resource is a compute instance. VIP: The backend resource is a virtual IP address. ELB: The backend resource is a load balancer. ELBv3: The backend resource is a dedicated load balancer. CUSTOMIZE: The backend resource is a user-defined IP address. Minimum: **1** Maximum: **10**
protocol	String	Specifies the protocol type. TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY are 6, 17, and 0, respectively. Minimum: **1** Maximum: **3** Enumeration values: • **tcp** • **udp** • **any**
private_ip_address	String	Specifies the private IP address of the backend instance. Minimum: **7** Maximum: **15**
internal_service_port	String	Specifies the port number of the backend instance. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
transit_service_port	String	Specifies the port number of the transit IP address. Minimum: **0** Maximum: **65535** Minimum: **1** Maximum: **5**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the DNAT rule when the DNAT rule is being created. Minimum: **1** Maximum: **36**

Parameter	Type	Description
created_at	String	Specifies when the DNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the DNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Example Requests

None

Example Responses

Status code: 200

DNAT rule details queried.

```
{
  "dnat_rule" : {
    "id" : "24dd6bf5-48f2-4915-ad0b-5bb111d39c83",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "description" : "aa",
    "gateway_id" : "0adefb29-a6c2-48a5-8637-2be67fa03fec",
    "transit_ip_id" : "3faa719d-6d18-4ccb-a5c7-33e65a09663e",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "network_interface_id" : "dae9393a-b536-491c-a5a2-72edc1104707",
    "type" : "COMPUTE",
    "protocol" : "any",
    "internal_service_port" : "0",
    "transit_service_port" : "0",
    "private_ip_address" : "192.168.1.72",
    "created_at" : "2019-04-29T07:10:01",
    "updated_at" : "2019-04-29T07:10:01"
  },
  "request_id" : "747a911c17067a39692f75ac146fb47e"
}
```

Status Codes

Status Code	Description
200	DNAT rule details queried.

Error Codes

See **Error Codes**.

6.3 SNAT Rules

6.3.1 Querying SNAT Rules

Function

This API is used to query SNAT rules.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/snat-rules

Table 6-59 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Table 6-60 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the number of records displayed on each page. The value ranges from 0 to 2000. Default value: 2000 Minimum: **1** Maximum: **2000** Default: **2000**

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the start resource ID of pagination query. If the parameter is left blank, only resources on the first page are queried. The value is obtained from next_marker or previous_marker in PageInfo queried last time. Minimum: **36** Maximum: **36**
page_reverse	No	Boolean	Specifies whether to query resources on the previous page.
id	No	Array	Specifies the SNAT rule ID.
project_id	No	Array	Specifies the project ID.
description	No	Array	Provides supplementary information about the SNAT rule.
gateway_id	No	Array	Specifies the private NAT gateway ID.
cidr	No	Array	Specifies the CIDR block that matches the SNAT rule.
virsubnet_id	No	Array	Specifies the ID of the subnet that matches the SNAT rule.
transit_ip_id	No	Array	Specifies the ID of the transit IP address.
transit_ip_address	No	Array	Specifies the transit IP address.
enterprise_project_id	No	Array	Specifies the ID of the enterprise project that is associated with the SNAT rule when the SNAT rule is being created.

Request Parameters

Table 6-61 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-62 Response body parameters

Parameter	Type	Description
snat_rules	Array of **PrivateSnat** objects	Specifies the response body for querying SNAT rules.
page_info	**PageInfo** object	Specifies the pagination information.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-63 PrivateSnat

Parameter	Type	Description
id	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**

Parameter	Type	Description
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
cidr	String	Specifies the CIDR block that matches the SNAT rule. Constraints: • Either this parameter or virsubnet_id must be specified. • The CIDR block cannot be the same as that of an existing SNAT rule. Minimum: **9** Maximum: **18**
virsubnet_id	String	Specifies the ID of the subnet that matches the SNAT rule. Constraint: Either this parameter or cidr must be specified. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the SNAT rule. Minimum: **1** Maximum: **36**
transit_ip_associations	Array of **AssociatedTransitIp** objects	Specifies the list of details of associated transit IP addresses.
created_at	String	Specifies when the SNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the SNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
enterprise_project_id	String	Specifies the enterprise project ID. Minimum: **1** Maximum: **36**

Table 6-64 AssociatedTransitIp

Parameter	Type	Description
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
transit_ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**

Table 6-65 PageInfo

Parameter	Type	Description
next_marker	String	Specifies the ID of the last record in this query, which can be used in the next query. Minimum: **1** Maximum: **36**
previous_marker	String	Specifies the ID of the first record in the pagination query result. When page_reverse is set to true, this parameter is used together to query resources on the previous page. Minimum: **1** Maximum: **36**
current_count	Integer	Specifies the ID of the last record in the pagination query result. It is usually used to query resources on the next page. Minimum: **1** Maximum: **2000**

Example Requests

None

Example Responses

Status code: 200

SNAT rules queried.

```
{
  "snat_rules" : [ {
    "id" : "8a522ff9-8158-494b-83cd-533b045700e6",
    "project_id" : "cfa563efb77d4b6d9960781d82530fd8",
    "description" : "snat rule description",
    "gateway_id" : "80da6f26-94eb-4537-97f0-5a56f4d04cfb",
```



```
  "cidr" : "",
  "virsubnet_id" : "95df1b88-d9bc-4edd-a808-a771dd4ded32",
  "transit_ip_associations" : [ {
    "transit_ip_id" : "bbe7c2e7-3bad-445b-a067-b30acce66053",
    "transit_ip_address" : "172.20.1.98"
  } ],
  "created_at" : "2019-10-22T03:33:07",
  "updated_at" : "2019-10-22T03:33:07"
 }, {
  "id" : "af4dbb83-7ca0-4ed1-b28b-668c1f9c6b81",
  "project_id" : "cfa563efb77d4b6d9960781d82530fd8",
  "description" : "snat rule description",
  "gateway_id" : "80da6f26-94eb-4537-97f0-5a56f4d04cfb",
  "cidr" : "",
  "virsubnet_id" : "5b9ea497-727d-4ad0-a99e-3984b3f5aaed",
  "transit_ip_associations" : [ {
    "transit_ip_id" : "36a3049a-1682-48b3-b1cf-cb986a3350ef",
    "transit_ip_address" : "172.20.1.10"
  } ],
  "created_at" : "2019-10-22T03:31:19",
  "updated_at" : "2019-10-22T03:31:19"
 } ],
 "page_info" : {
  "next_marker" : "af4dbb83-7ca0-4ed1-b28b-668c1f9c6b81",
  "previous_marker" : "8a522ff9-8158-494b-83cd-533b045700e6",
  "current_count" : 2
 },
 "request_id" : "69806207-62e3-4950-b463-ff5c1779b714"
}
```


Status Codes

Status Code	Description
200	SNAT rules queried.

Error Codes

See **Error Codes**.

6.3.2 Querying Details About a Specified SNAT Rule

Function

This API is used to query details about a specified SNAT rule.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/snat-rules/{snat_rule_id}

Table 6-66 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
snat_rule_id	Yes	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-67 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-68 Response body parameters

Parameter	Type	Description
snat_rule	**PrivateSnat** object	Specifies the response body of the SNAT rule.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-69 PrivateSnat

Parameter	Type	Description
id	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
cidr	String	Specifies the CIDR block that matches the SNAT rule. Constraints: • Either this parameter or virsubnet_id must be specified. • The CIDR block cannot be the same as that of an existing SNAT rule. Minimum: **9** Maximum: **18**
virsubnet_id	String	Specifies the ID of the subnet that matches the SNAT rule. Constraint: Either this parameter or cidr must be specified. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the SNAT rule. Minimum: **1** Maximum: **36**
transit_ip_associations	Array of **AssociatedTransitIp** objects	Specifies the list of details of associated transit IP addresses.
created_at	String	Specifies when the SNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Parameter	Type	Description
updated_at	String	Specifies when the SNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
enterprise_project_id	String	Specifies the enterprise project ID. Minimum: **1** Maximum: **36**

Table 6-70 AssociatedTransitIp

Parameter	Type	Description
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
transit_ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**

Example Requests

None

Example Responses

Status code: 200

SNAT rule details queried.

```
{
  "snat_rule" : {
    "id" : "8a522ff9-8158-494b-83cd-533b045700e6",
    "project_id" : "cfa563efb77d4b6d9960781d82530fd8",
    "description" : "my_snat_rule02",
    "gateway_id" : "80da6f26-94eb-4537-97f0-5a56f4d04cfb",
    "cidr" : "",
    "virsubnet_id" : "95df1b88-d9bc-4edd-a808-a771dd4ded32",
    "transit_ip_associations" : [ {
      "transit_ip_id" : "bbe7c2e7-3bad-445b-a067-b30acce66053",
      "transit_ip_address" : "172.20.1.98"
    } ],
    "created_at" : "2019-10-22T03:33:07",
    "updated_at" : "2019-10-22T03:33:07"
  },
  "request_id" : "c8b21002-a594-414d-9585-2cc5963d4c3e"
}
```


Status Codes

Status Code	Description
200	SNAT rule details queried.

Error Codes

See **Error Codes**.

6.3.3 Updating an SNAT Rule

Function

This API is used to update a specified SNAT rule.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

PUT /v3/{project_id}/private-nat/snat-rules/{snat_rule_id}

Table 6-71 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
snat_rule_id	Yes	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-72 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Table 6-73 Request body parameters

Parameter	Mandatory	Type	Description
snat_rule	Yes	**UpdatePrivateSnatOption** object	Specifies the request body for updating an SNAT rule.

Table 6-74 UpdatePrivateSnatOption

Parameter	Mandatory	Type	Description
transit_ip_ids	No	Array of strings	Specifies the ID list of transit IP addresses. Minimum: **36** Maximum: **36**
description	No	String	Provides supplementary information about the SNAT rule. Minimum: **1** Maximum: **36**

Response Parameters

Status code: 200

Table 6-75 Response body parameters

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **36** Maximum: **36**
snat_rule	**PrivateSnat** object	Specifies the response body of the SNAT rule.

Table 6-76 PrivateSnat

Parameter	Type	Description
id	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
cidr	String	Specifies the CIDR block that matches the SNAT rule. Constraints: • Either this parameter or virsubnet_id must be specified. • The CIDR block cannot be the same as that of an existing SNAT rule. Minimum: **9** Maximum: **18**
virsubnet_id	String	Specifies the ID of the subnet that matches the SNAT rule. Constraint: Either this parameter or cidr must be specified. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the SNAT rule. Minimum: **1** Maximum: **36**

Parameter	Type	Description
transit_ip_associations	Array of **AssociatedTransitIp** objects	Specifies the list of details of associated transit IP addresses.
created_at	String	Specifies when the SNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the SNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
enterprise_project_id	String	Specifies the enterprise project ID. Minimum: **1** Maximum: **36**

Table 6-77 AssociatedTransitIp

Parameter	Type	Description
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
transit_ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**

Example Requests

Specifies the request body for updating an SNAT rule.

```
 https://{Endpoint}/v3/ cfa563efb77d4b6d9960781d82530fd8/private-nat/snat-rules/aef81f7c-1be5-4aad-b6e9-23b2ec3bc395

{
  "snat_rule" : {
    "description" : "my_snat_rule_update",
    "transit_ip_ids" : [ "bbe7c2e7-3bad-445b-a067-b30acce66053" ]
  }
}
```


Example Responses

Status code: 200

SNAT rule updated.

```
{
  "request_id" : "15bd32b2-1464-4817-b559-444d22499f6c",
  "snat_rule" : {
    "id" : "af4dbb83-7ca0-4ed1-b28b-668c1f9c6b81",
    "project_id" : "cfa563efb77d4b6d9960781d82530fd8",
    "description" : "my_snat_rule_update",
    "gateway_id" : "80da6f26-94eb-4537-97f0-5a56f4d04cfb",
    "cidr" : "10.1.1.64/30",
    "virsubnet_id" : "",
    "transit_ip_associations" : [ {
      "transit_ip_id" : "bbe7c2e7-3bad-445b-a067-b30acce66053",
      "transit_ip_address" : "172.20.1.98"
    } ],
    "created_at" : "2019-10-22T03:31:19",
    "updated_at" : "2019-10-22T03:39:52"
  }
}
```

Status Codes

Status Code	Description
200	SNAT rule updated.

Error Codes

See **Error Codes**.

6.3.4 Creating an SNAT Rule

Function

This API is used to create an SNAT rule.

Constraints

When you are creating an SNAT rule, status of the NAT gateway must be set to ACTIVE.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat/snat-rules

Table 6-78 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Request Parameters

Table 6-79 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Table 6-80 Request body parameters

Parameter	Mandatory	Type	Description
snat_rule	Yes	**CreatePrivateSnatOption** object	Specifies the request body for creating an SNAT rule.

Table 6-81 CreatePrivateSnatOption

Parameter	Mandatory	Type	Description
gateway_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**

Parameter	Mandatory	Type	Description
cidr	No	String	Specifies the CIDR block that matches the SNAT rule. Constraint: Either this parameter or virsubnet_id must be specified. Minimum: **9** Maximum: **18**
virsubnet_id	No	String	Specifies the ID of the subnet that matches the SNAT rule. Constraint: Either this parameter or cidr must be specified. Minimum: **36** Maximum: **36**
description	No	String	Provides supplementary information about the SNAT rule. Minimum: **0** Maximum: **255**
transit_ip_ids	Yes	Array of strings	Specifies the ID list of transit IP addresses. Minimum: **36** Maximum: **36**

Response Parameters

Status code: 201

Table 6-82 Response body parameters

Parameter	Type	Description
snat_rule	**PrivateSnat** object	Specifies the response body of the SNAT rule.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-83 PrivateSnat

Parameter	Type	Description
id	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **36** Maximum: **36**
gateway_id	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**
cidr	String	Specifies the CIDR block that matches the SNAT rule. Constraints: • Either this parameter or virsubnet_id must be specified. • The CIDR block cannot be the same as that of an existing SNAT rule. Minimum: **9** Maximum: **18**
virsubnet_id	String	Specifies the ID of the subnet that matches the SNAT rule. Constraint: Either this parameter or cidr must be specified. Minimum: **36** Maximum: **36**
description	String	Provides supplementary information about the SNAT rule. Minimum: **1** Maximum: **36**
transit_ip_associations	Array of **AssociatedTransitIp** objects	Specifies the list of details of associated transit IP addresses.
created_at	String	Specifies when the SNAT rule was created. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Parameter	Type	Description
updated_at	String	Specifies when the SNAT rule was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
enterprise_project_id	String	Specifies the enterprise project ID. Minimum: **1** Maximum: **36**

Table 6-84 AssociatedTransitIp

Parameter	Type	Description
transit_ip_id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
transit_ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**

Example Requests

Specifies the request body for creating an SNAT rule.

```
https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat/snat-rules

{
  "snat_rule" : {
    "description" : "my_snat_rule01",
    "gateway_id" : "80da6f26-94eb-4537-97f0-5a56f4d04cfb",
    "virsubnet_id" : "5b9ea497-727d-4ad0-a99e-3984b3f5aaed",
    "transit_ip_ids" : [ "36a3049a-1682-48b3-b1cf-cb986a3350ef" ]
  }
}
```

Example Responses

Status code: 201

SNAT rule created.

```
{
  "snat_rule" : {
    "id" : "af4dbb83-7ca0-4ed1-b28b-668c1f9c6b81",
    "project_id" : "cfa563efb77d4b6d9960781d82530fd8",
    "description" : "snat rule description",
    "gateway_id" : "80da6f26-94eb-4537-97f0-5a56f4d04cfb",
    "cidr" : "",
    "virsubnet_id" : "5b9ea497-727d-4ad0-a99e-3984b3f5aaed",
```



```
  "transit_ip_associations" : [ {
    "transit_ip_id" : "36a3049a-1682-48b3-b1cf-cb986a3350ef",
    "transit_ip_address" : "172.20.1.10"
  } ],
  "created_at" : "2019-10-22T03:31:19",
  "updated_at" : "2019-10-22T03:31:19"
 },
 "request_id" : "2937502e-73f9-4ba5-ae75-2293a0b35fb8"
}
```


Status Codes

Status Code	Description
201	SNAT rule created.

Error Codes

See **Error Codes**.

6.3.5 Deleting an SNAT Rule

Function

This API is used to delete a specified SNAT rule.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

DELETE /v3/{project_id}/private-nat/snat-rules/{snat_rule_id}

Table 6-85 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
snat_rule_id	Yes	String	Specifies the SNAT rule ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-86 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

None

Example Requests

None

Example Responses

None

Status Codes

Status Code	Description
204	SNAT rule deleted.

Error Codes

See **Error Codes**.

6.4 Transit IP Addresses

6.4.1 Querying Transit IP Addresses

Function

This API is to query transit IP addresses.

Constraints

You can type the question mark (?) and ampersand (&) at the end of the URI to define multiple search criteria. All optional parameters can be filtered. For details, see the example request.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/transit-ips

Table 6-87 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Table 6-88 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the number of records displayed on each page. The value ranges from 0 to 2000. Default value: 2000 Minimum: **1** Maximum: **2000** Default: **2000**
marker	No	String	Specifies the start resource ID of pagination query. If the parameter is left blank, only resources on the first page are queried. The value is obtained from next_marker or previous_marker in PageInfo queried last time. Minimum: **36** Maximum: **36**
page_reverse	No	Boolean	Specifies whether to query resources on the previous page.

Parameter	Mandatory	Type	Description
id	No	Array	Specifies the ID of the transit IP address.
project_id	No	Array	Specifies the project ID.
network_interface_id	No	Array	Specifies the network interface ID of the transit IP address.
ip_address	No	Array	Specifies the transit IP address.
gateway_id	No	Array	Specifies the ID of the private NAT gateway associated with the transit IP address.
enterprise_project_id	No	Array	Specifies the ID of the enterprise project that is associated with the transit IP address when the transit IP address is being assigned.
virsubnet_id	No	Array	Specifies the subnet ID of the current tenant.

Request Parameters

Table 6-89 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-90 Response body parameters

Parameter	Type	Description
transit_ips	Array of **TransitIp** objects	Specifies the response body for querying transit IP addresses.
page_info	**PageInfo** object	Specifies the pagination information.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-91 TransitIp

Parameter	Type	Description
id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **1** Maximum: **36**
network_interface_id	String	Specifies the network interface ID of the transit IP address. Minimum: **36** Maximum: **36**
ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**
created_at	String	Specifies when the transit IP address was assigned. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the transit IP address was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**

Parameter	Type	Description
virsubnet_id	String	Specifies the subnet ID of the current tenant. Minimum: **0** Maximum: **36**
tags	Array of **Tag** objects	Specifies the list of tags.
gateway_id	String	Specifies the ID of the private NAT gateway associated with the transit IP address. Minimum: **36** Maximum: **36**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the transit IP address when the transit IP address is being assigned. Minimum: **1** Maximum: **36**

Table 6-92 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Table 6-93 PageInfo

Parameter	Type	Description
next_marker	String	Specifies the ID of the last record in this query, which can be used in the next query. Minimum: **1** Maximum: **36**

Parameter	Type	Description
previous_marker	String	Specifies the ID of the first record in the pagination query result. When page_reverse is set to true, this parameter is used together to query resources on the previous page. Minimum: **1** Maximum: **36**
current_count	Integer	Specifies the ID of the last record in the pagination query result. It is usually used to query resources on the next page. Minimum: **1** Maximum: **2000**

Example Requests

None

Example Responses

Status code: 200

Transit IP addresses queried.


```
{
  "transit_ips" : [ {
    "id" : "3faa719d-6d18-4ccb-a5c7-33e65a09663e",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "network_interface_id" : "c91c43fb-8d66-48df-bfa9-b89053ac3737",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "gateway_id" : "521bb3d9-8bed-4c6c-9ee8-669bd0620f76",
    "ip_address" : "192.168.1.68",
    "created_at" : "2019-04-29T03:41:59",
    "updated_at" : "2019-04-29T03:41:59",
    "virsubnet_id" : "49ee5fb5-75bf-4320-946e-b21ef4c9c9c1",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ]
  }, {
    "id" : "a2845109-3b2f-4627-b08f-09a726c0a6e7",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "network_interface_id" : "adebbdca-8c26-4c14-b34f-3f53cd2c42f2",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "gateway_id" : "521bb3d9-8bed-4c6c-9ee8-669bd0620f76",
    "ip_address" : "192.168.1.58",
    "created_at" : "2019-04-29T02:16:09",
    "updated_at" : "2019-04-29T02:16:09",
    "virsubnet_id" : "333e5fb5-75bf-4320-946e-b21ef4c9c2g5",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ]
  } ],
  "request_id" : "747a911c17067a39692f75ac146fb47e"
}
```


Status Codes

Status Code	Description
200	Transit IP addresses queried.

Error Codes

See **Error Codes**.

6.4.2 Releasing a Transit IP Address

Function

This API is used to release a transit IP address.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

DELETE /v3/{project_id}/private-nat/transit-ips/{transit_ip_id}

Table 6-94 Path Parameters

Parameter	Mandatory	Type	Description
transit_ip_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **1** Maximum: **36**
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Request Parameters

Table 6-95 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **512**

Response Parameters

None

Example Requests

None

Example Responses

None

Status Codes

Status Code	Description
204	Transit IP address released.

Error Codes

See **Error Codes**.

6.4.3 Assigning a Transit IP Address

Function

This API is used to assign a transit IP address.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat/transit-ips

Table 6-96 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Request Parameters

Table 6-97 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **512**

Table 6-98 Request body parameters

Parameter	Mandatory	Type	Description
transit_ip	Yes	**CreatTransitIpOption** object	Specifies the request body for assigning a transit IP address.

Table 6-99 CreatTransitIpOption

Parameter	Mandatory	Type	Description
virsubnet_id	Yes	String	Specifies the subnet ID of the current project. Minimum: **1** Maximum: **36**
ip_address	No	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**
enterprise_project_id	No	String	Specifies the ID of the enterprise project that is associated with the transit IP address when the transit IP address is being assigned. Default: **0** Minimum: **1** Maximum: **36**
tags	No	Array of **Tag** objects	Specifies the tag of the transit IP address.

Table 6-100 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Response Parameters

Status code: 201

Table 6-101 Response body parameters

Parameter	Type	Description
transit_ip	**TransitIp** object	Specifies the response body of the transit IP address.

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-102 TransitIp

Parameter	Type	Description
id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **1** Maximum: **36**
network_interface_id	String	Specifies the network interface ID of the transit IP address. Minimum: **36** Maximum: **36**
ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**
created_at	String	Specifies when the transit IP address was assigned. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the transit IP address was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
virsubnet_id	String	Specifies the subnet ID of the current tenant. Minimum: **0** Maximum: **36**
tags	Array of **Tag** objects	Specifies the list of tags.

Parameter	Type	Description
gateway_id	String	Specifies the ID of the private NAT gateway associated with the transit IP address. Minimum: **36** Maximum: **36**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the transit IP address when the transit IP address is being assigned. Minimum: **1** Maximum: **36**

Table 6-103 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

Assigning a transit IP address

```
POST https://{Endpoint}/v3/70505c941b9b4dfd82fd351932328a2f/private-nat/transit-ips

{
  "transit_ip" : {
    "virsubnet_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "ip_address" : "192.168.1.68",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ]
  }
}
```

Example Responses

Status code: 201

Transit IP address assigned.

```
{
  "transit_ip" : {
    "id" : "a2845109-3b2f-4627-b08f-09a726c0a6e7",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
```



```
  "network_interface_id" : "adebbdca-8c26-4c14-b34f-3f53cd2c42f2",
  "ip_address" : "192.168.1.58",
  "gateway_id" : "521bb3d9-8bed-4c6c-9ee8-669bd0620f76",
  "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
  "created_at" : "2019-04-29T02:16:09",
  "updated_at" : "2019-04-29T02:16:09",
  "virsubnet_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  } ]
 },
 "request_id" : "747a911c17067a39692f75ac146fb47e"
}
```


Status Codes

Status Code	Description
201	Transit IP address assigned.

Error Codes

See **Error Codes**.

6.4.4 Querying Details About a Specified Transit IP Address.

Function

This API is used to query details about a specified transit IP address.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat/transit-ips/{transit_ip_id}

Table 6-104 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **36**
transit_ip_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-105 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token value. Minimum: **1** Maximum: **512**

Response Parameters

Status code: 200

Table 6-106 Response body parameters

Parameter	Type	Description
transit_ip	**TransitIp** object	Specifies the response body of the transit IP address.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**

Table 6-107 TransitIp

Parameter	Type	Description
id	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**
project_id	String	Specifies the project ID. Minimum: **1** Maximum: **36**

Parameter	Type	Description
network_interface_id	String	Specifies the network interface ID of the transit IP address. Minimum: **36** Maximum: **36**
ip_address	String	Specifies the transit IP address. Minimum: **7** Maximum: **35**
created_at	String	Specifies when the transit IP address was assigned. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
updated_at	String	Specifies when the transit IP address was updated. It is a UTC time in the yyyy-mm-ddThh:mm:ssZ format. Minimum: **1** Maximum: **36**
virsubnet_id	String	Specifies the subnet ID of the current tenant. Minimum: **0** Maximum: **36**
tags	Array of **Tag** objects	Specifies the list of tags.
gateway_id	String	Specifies the ID of the private NAT gateway associated with the transit IP address. Minimum: **36** Maximum: **36**
enterprise_project_id	String	Specifies the ID of the enterprise project that is associated with the transit IP address when the transit IP address is being assigned. Minimum: **1** Maximum: **36**

Table 6-108 Tag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**

Parameter	Type	Description
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

None

Example Responses

Status code: 200

Details about a specified transit IP address queried.


```
{
  "transit_ip" : {
    "id" : "a2845109-3b2f-4627-b08f-09a726c0a6e7",
    "project_id" : "da261828016849188f4dcc2ef94d9da9",
    "network_interface_id" : "adebbdca-8c26-4c14-b34f-3f53cd2c42f2",
    "ip_address" : "192.168.1.58",
    "gateway_id" : "521bb3d9-8bed-4c6c-9ee8-669bd0620f76",
    "enterprise_project_id" : "2759da7b-8015-404c-ae0a-a389007b0e2a",
    "created_at" : "2019-04-29T02:16:09",
    "updated_at" : "2019-04-29T02:16:09",
    "virsubnet_id" : "49ee5fb5-75bf-4320-946e-b21ef4c9c9c1",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    } ]
  },
  "request_id" : "747a911c17067a39692f75ac146fb47e"
}
```


Status Codes

Status Code	Description
200	Details about a specified transit IP address queried.

Error Codes

See **Error Codes**.

6.5 Private NAT Gateway Tags

6.5.1 Querying Private NAT Gateways

Function

- This API is used to query private NAT gateways by tag.
- TMS uses this API to query and list private NAT gateways by tag.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat-gateways/resource_instances/action

Table 6-109 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**

Request Parameters

Table 6-110 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Table 6-111 Request body parameters

Parameter	Mandatory	Type	Description
offset	No	String	Specifies the request ID. Minimum: **0** Maximum: **65535**
limit	No	String	Tags Minimum: **1** Maximum: **1000**
action	Yes	String	Specifies the operation to perform, which can only be filter (filtering) or count (querying the total number). filter indicates pagination query. count indicates that the total number of query results meeting the search criteria will be returned. Enumeration values: • **filter** • **count**
matches	No	Array of **Match** objects	Tags
not_tags	No	Array of **Tags** objects	Specifies details of a resource. This parameter is used for extension and is left blank by default.
tags	No	Array of **Tags** objects	Specifies the tag key.

Parameter	Mandatory	Type	Description
tags_any	No	Array of **Tags** objects	Specifies any tag that is included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned.
not_tags_any	No	Array of **Tags** objects	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned.

Table 6-112 Match

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key used to search resources. Minimum: **1** Maximum: **128**

Parameter	Mandatory	Type	Description
value	Yes	String	Specifies the tag value used to search resources. Minimum: **0** Maximum: **255**

Table 6-113 Tags

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the operation to perform, which can only be filter (filtering) or count (querying the total number).filter indicates pagination query. count indicates that the total number of query results meeting the search criteria will be returned. Minimum: **1** Maximum: **128**
values	Yes	Array of strings	Specifies any tag that is included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **0** Maximum: **255**

Response Parameters

Status code: 200

Table 6-114 Response body parameters

Parameter	Type	Description
resources	Array of **Resource** objects	Specifies the resource list.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
total_count	Integer	Specifies the total number of records.

Table 6-115 Resource

Parameter	Type	Description
resource_detail	Object	Specifies details of a resource. This parameter is used for extension and is left blank by default.
resource_id	String	Specifies the resource ID. Minimum: **36** Maximum: **36**
resource_name	String	Specifies the resource name. This parameter is an empty string by default if there is no resource name. Minimum: **0** Maximum: **36**
resource_tag	Array of **ResourceTag** objects	Specifies the list of queried tags. If no tag is matched, an empty array is returned.

Table 6-116 ResourceTag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

- The request body when action is set to filter

```
POST  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/
resource_instances/action

{
  "offset" : "10",
  "limit" : "10",
  "action" : "filter",
  "matches" : [ {
    "key" : "resource_name",
    "value" : "resource1"
  } ],
  "not_tags" : [ {
    "key" : "key1",
    "values" : [ "*value1", "value2" ]
  } ],
  "tags" : [ {
    "key" : "key1",
    "values" : [ "*value1", "value2" ]
  } ],
  "tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "not_tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ]
}
```

- The request body when action is set to count

```
POST  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/
resource_instances/action

{
  "action" : "count",
  "not_tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "*value2" ]
  } ],
  "tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  }, {
    "key" : "key2",
    "values" : [ "value1", "value2" ]
  } ],
  "tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "not_tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "matches" : [ {
    "key" : "resource_name",
    "value" : "resource1"
  } ]
}
```

Example Responses

Status code: 200

- Query operation succeeded.
- Example 1: the response body when action is set to count
- Example 2: the response body when action is set to filter
- Example 1

```
{
  "request_id" : "a67262f6b7242d63d4ae95e41abf2790",
  "total_count" : 100
}
```

- Example 2

```
{
  "resources" : [ {
    "resource_detail" : null,
    "resource_id" : "e5ad289f-9c56-4daf-b08b-2e53a983473a",
    "resource_name" : "nat_gateways",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    }, {
      "key" : "key2",
      "value" : "value1"
    } ]
  } ],
  "request_id" : "a67262f6b7242d63d4ae95e41abf2790",
  "total_count" : 1
}
```

Status Codes

Status Code	Description
200	• Query operation succeeded. • Example 1: the response body when action is set to count • Example 2: the response body when action is set to filter

Error Codes

See **Error Codes**.

6.5.2 Querying Tags of All Private NAT Gateways in a Project

Function

- This API is used to query tags of all private NAT gateways owned by a tenant in a specified project.
- TMS uses this API to list tags of all private NAT gateways owned by a tenant, and to provide tag association when users tag or query private NAT gateway.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat-gateways/tags

Table 6-117 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**

Request Parameters

Table 6-118 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-119 Response body parameters

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
tags	Array of **Tags** objects	Tags

Table 6-120 Tags

Parameter	Type	Description
key	String	Specifies the operation to perform, which can only be filter (filtering) or count (querying the total number).filter indicates pagination query. count indicates that the total number of query results meeting the search criteria will be returned. Minimum: **1** Maximum: **128**
values	Array of strings	Specifies any tag that is included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **0** Maximum: **255**

Example Requests

```
GET https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/tags
```

Example Responses

Status code: 200

Query operation succeeded.

```
{
  "request_id" : "c285190c-b9e9-4f38-a69a-6745f22d8dca",
  "tags" : [ {
    "key" : "keys",
    "values" : [ "value" ]
  }, {
    "key" : "key3",
    "values" : [ "value3", "value33" ]
  }, {
    "key" : "key1",
    "values" : [ "value1" ]
  }, {
    "key" : "key2",
    "values" : [ "value2", "value22" ]
  } ]
}
```


Status Codes

Status Code	Description
200	Query operation succeeded.

Error Codes

See **Error Codes**.

6.5.3 Querying Tags of a Private NAT Gateway

Function

- This API is used to query tags of a specified private NAT gateway.
- TMS uses this API to query all tags of a specified private NAT gateway.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/private-nat-gateways/{resource_id}/tags

Table 6-121 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-122 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-123 Response body parameters

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
tags	Array of **Tag** objects	Tags

Table 6-124 Tag

Parameter	Type	Description
key	String	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **1** Maximum: **128**

Parameter	Type	Description
value	String	Specifies the request ID. Minimum: **0** Maximum: **255**

Example Requests

```
GET  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/b0399473-c352-4d0c-8ff4-cfde719cfde9/tags
```

Example Responses

Status code: 200

Query operation succeeded.

```
{
  "request_id" : "80ef5f21-b81a-4546-b23d-84272507d330",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value2"
  }, {
    "key" : "key3",
    "value" : "value3"
  } ]
}
```

Status Codes

Status Code	Description
200	Query operation succeeded.

Error Codes

See **Error Codes**.

6.5.4 Adding a Tag to a Private NAT Gateway

Function

- A private NAT gateway can have up to 10 tags.
- This API is idempotent.
- If a tag to be created has the same key as an existing tag, the tag will be created and overwrite the existing one.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat-gateways/{resource_id}/tags

Table 6-125 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-126 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Table 6-127 Request body parameters

Parameter	Mandatory	Type	Description
tag	Yes	**Tag** object	Specifies tags.

Table 6-128 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the request ID. Minimum: **0** Maximum: **255**

Response Parameters

None

Example Requests

```
POST  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/3320166e-b937-40cc-a35c-02cd3f2b3ee2/tags

{
  "tag" : {
    "key" : "key1",
    "value" : "value1"
  }
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Tag added.

Error Codes

See **Error Codes**.

6.5.5 Batch Adding Tags to or Deleting Tags from a Private NAT Gateway

Function

- This API is used to batch add tags to or delete tags from a private NAT gateway.
- TMS uses this API to batch manage tags of a private NAT gateway.
- A private NAT gateway can have up to 10 tags.

Constraints

This API is idempotent.

- If the request body contains duplicate keys, an error is reported.
- During tag creation, duplicate keys are not allowed. If a key already exists in the database, its value will be overwritten by the new duplicate key.
- During tag deletion, if some tags to be deleted do not exist, the operation is considered to be successful by default. The character set of the tags will not be verified.
- The tags structure cannot be missing during deletion. The key cannot be left blank or be an empty string.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/private-nat-gateways/{resource_id}/tags/action

Table 6-129 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the private NAT gateway ID. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-130 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Table 6-131 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the operation to perform. The value can be: create delete Enumeration values: • **create** • **delete**
tags	Yes	Array of **Tag** objects	Specifies the tags.

Table 6-132 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the request ID. Minimum: **0** Maximum: **255**

Response Parameters

None

Example Requests

- Batch adding tags to a private NAT gateway

```
POST  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/3320166e-b937-40cc-a35c-02cd3f2b3ee2/tags/action

{
  "action" : "create",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value2"
  } ]
}
```

- Batch deleting tags from a private NAT gateway

```
POST  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/3320166e-b937-40cc-a35c-02cd3f2b3ee2/tags/action

{
  "action" : "delete",
  "tags" : [ {
```



```
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value2"
  } ]
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Tags added or deleted.

Error Codes

See **Error Codes**.

6.5.6 Deleting a Tag from a Private NAT Gateway

Function

- This API is idempotent.
- When a tag is deleted, the tag character set is not verified. Before calling this API, encode the API URL. If the key of the tag to be deleted does not exist, 404 is displayed. The tag key cannot be left blank or be an empty string.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

DELETE /v3/{project_id}/private-nat-gateways/{resource_id}/tags/{key}

Table 6-133 Path Parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. Minimum: **1** Maximum: **128**

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the ID of the private NAT gateway. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-134 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Response Parameters

None

Example Requests

Deleting a tag from a private NAT gateway

```
DELETE https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/private-nat-gateways/3320166e-b937-40cc-a35c-02cd3f2b3ee2/tags/key1
```

Example Responses

None

Status Codes

Status Code	Description
204	Tag deleted.

Error Codes

See **Error Codes**.

6.6 Transit IP Address Tags

6.6.1 Querying Transit IP Addresses

Function

- This API is used to query transit IP addresses by tag.
- TMS uses this API to query and list transit IP addresses by tag.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/transit-ips/resource_instances/action

Table 6-135 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**

Request Parameters

Table 6-136 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Table 6-137 Request body parameters

Parameter	Mandatory	Type	Description
offset	No	String	Specifies the request ID. Minimum: **0** Maximum: **65535**
limit	No	String	Tags Minimum: **1** Maximum: **1000**
action	Yes	String	Specifies the operation to perform, which can only be filter (filtering) or count (querying the total number). filter indicates pagination query. count indicates that the total number of query results meeting the search criteria will be returned. Enumeration values: • **filter** • **count**
matches	No	Array of **Match** objects	Tags

Parameter	Mandatory	Type	Description
not_tags	No	Array of **Tags** objects	Specifies details of a resource. This parameter is used for extension and is left blank by default.
tags	No	Array of **Tags** objects	Specifies the tag key.
tags_any	No	Array of **Tags** objects	Specifies any tag that is included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned.
not_tags_any	No	Array of **Tags** objects	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned.

Table 6-138 Match

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key used to search resources. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the tag value used to search resources. Minimum: **0** Maximum: **255**

Table 6-139 Tags

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the operation to perform, which can only be filter (filtering) or count (querying the total number).filter indicates pagination query. count indicates that the total number of query results meeting the search criteria will be returned. Minimum: **1** Maximum: **128**
values	Yes	Array of strings	Specifies any tag that is included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **0** Maximum: **255**

Response Parameters

Status code: 200

Table 6-140 Response body parameters

Parameter	Type	Description
resources	Array of **Resource** objects	Specifies the resource list.
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
total_count	Integer	Specifies the total number of records.

Table 6-141 Resource

Parameter	Type	Description
resource_detail	Object	Specifies details of a resource. This parameter is used for extension and is left blank by default.
resource_id	String	Specifies the resource ID. Minimum: **36** Maximum: **36**
resource_name	String	Specifies the resource name. This parameter is an empty string by default if there is no resource name. Minimum: **0** Maximum: **36**
resource_tag	Array of **ResourceTag** objects	Specifies the list of queried tags. If no tag is matched, an empty array is returned.

Table 6-142 ResourceTag

Parameter	Type	Description
key	String	Specifies the tag key. Minimum: **1** Maximum: **128**
value	String	Specifies the tag value. Minimum: **0** Maximum: **255**

Example Requests

- The request body when action is set to filter

```
POST https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/resource_instances/action

{
  "offset" : "10",
  "limit" : "10",
  "action" : "filter",
  "matches" : [ {
    "key" : "resource_name",
    "value" : "resource1"
  } ],
  "not_tags" : [ {
    "key" : "key1",
    "values" : [ "*value1", "value2" ]
  } ],
  "tags" : [ {
    "key" : "key1",
    "values" : [ "*value1", "value2" ]
  } ],
  "tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "not_tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ]
}
```

- The request body when action is set to count

```
POST https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/resource_instances/action

{
  "action" : "count",
  "not_tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "*value2" ]
  } ],
  "tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  }, {
    "key" : "key2",
    "values" : [ "value1", "value2" ]
  } ],
  "tags_any" : [ {
```



```
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "not_tags_any" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ],
  "matches" : [ {
    "key" : "resource_name",
    "value" : "resource1"
  } ]
}
```

Example Responses

Status code: 200

- Query operation succeeded.
- Example 1: the response body when action is set to count
- Example 2: the response body when action is set to filter
- Example 1

```
{
  "request_id" : "d70aabc854d3d301f9bb106e6b70ac99",
  "total_count" : 100
}
```

- Example 2

```
{
  "resources" : [ {
    "resource_detail" : null,
    "resource_id" : "ae33be9b-d2c0-441b-a8d0-f6dafedf1778",
    "resource_name" : "transit_ips",
    "tags" : [ {
      "key" : "key1",
      "value" : "value1"
    }, {
      "key" : "key2",
      "value" : "value1"
    } ]
  } ],
  "request_id" : "9e47d9476cfd346f864cb77acb274185",
  "total_count" : 1
}
```

Status Codes

Status Code	Description
200	• Query operation succeeded. • Example 1: the response body when action is set to count • Example 2: the response body when action is set to filter

Error Codes

See **Error Codes**.

6.6.2 Querying Tags of All Transit IP Addresses in a Specified Project

Function

- This API is used to query tags of all transit IP addresses owned by a tenant in a specified project.
- TMS uses this API to list tags of all transit IP addresses owned by a tenant, and to provide tag association when users tag or query transit IP addresses.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/transit-ips/tags

Table 6-143 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**

Request Parameters

Table 6-144 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-145 Response body parameters

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
tags	Array of **Tags** objects	Tags

Table 6-146 Tags

Parameter	Type	Description
key	String	Specifies the operation to perform, which can only be filter (filtering) or count (querying the total number).filter indicates pagination query. count indicates that the total number of query results meeting the search criteria will be returned. Minimum: **1** Maximum: **128**
values	Array of strings	Specifies any tag that is included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **0** Maximum: **255**

Example Requests

```
GET  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/tags
```

Example Responses

Status code: 200

Query operation succeeded.

```
{
  "request_id" : "36479272a29de0be0a8a8b294c02ab7a",
  "tags" : [ {
    "key" : "keys",
```



```
    "values" : [ "value" ]
  }, {
    "key" : "key3",
    "values" : [ "value3", "value33" ]
  }, {
    "key" : "key1",
    "values" : [ "value1" ]
  }, {
    "key" : "key2",
    "values" : [ "value2", "value22" ]
  } ]
}
```

Status Codes

Status Code	Description
200	Query operation succeeded.

Error Codes

See **Error Codes**.

6.6.3 Querying Tags of a Transit IP Address

Function

- This API is used to query tags of a specified transit IP address.
- TMS uses this API to query all tags of a specified transit IP address.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

GET /v3/{project_id}/transit-ips/{resource_id}/tags

Table 6-147 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-148 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Response Parameters

Status code: 200

Table 6-149 Response body parameters

Parameter	Type	Description
request_id	String	Specifies the request ID. Minimum: **1** Maximum: **36**
tags	Array of **Tag** objects	Tags

Table 6-150 Tag

Parameter	Type	Description
key	String	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **1** Maximum: **128**

Parameter	Type	Description
value	String	Specifies the request ID. Minimum: **0** Maximum: **255**

Example Requests

```
GET  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/b0399473-c352-4d0c-8ff4-cfde719cfde9/tags
```

Example Responses

Status code: 200

Query operation succeeded.

```
{
  "request_id" : "80ef5f21-b81a-4546-b23d-84272507d330",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value2"
  }, {
    "key" : "key3",
    "value" : "value3"
  } ]
}
```

Status Codes

Status Code	Description
200	Query operation succeeded.

Error Codes

See **Error Codes**.

6.6.4 Adding a Tag to a Transit IP Address

Function

- A transit IP address can have up to 10 tags.
- This API is idempotent.
- If a tag to be created has the same key as an existing tag, the tag will be created and overwrite the existing one.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/transit-ips/{resource_id}/tags

Table 6-151 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-152 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Table 6-153 Request body parameters

Parameter	Mandatory	Type	Description
tag	Yes	**Tag** object	Specifies tags.

Table 6-154 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the request ID. Minimum: **0** Maximum: **255**

Response Parameters

None

Example Requests

```
POST  https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/56121618-fb0a-4a51-aff0-e2eb9cba4c73/tags

{
  "tag" : {
    "key" : "key1",
    "value" : "value1"
  }
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Tag added.

Error Codes

See **Error Codes**.

6.6.5 Batch Adding Tags to or Deleting Tags from a Transit IP Address

Function

- This API is used to batch add tags to or delete tags from a specified transit IP address.
- TMS needs to use this API to batch manage tags of a specified transit IP address.
- A transit IP address can have up to 10 tags.

Constraints

This API is idempotent.

- If the request body contains duplicate keys, an error is reported.
- During tag creation, duplicate keys are not allowed. If a key already exists in the database, its value will be overwritten by the new duplicate key.
- During tag deletion, if some tags to be deleted do not exist, the operation is considered to be successful by default. The character set of the tags will not be verified.
- The tags structure cannot be missing during deletion. The key cannot be left blank or be an empty string.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

POST /v3/{project_id}/transit-ips/{resource_id}/tags/action

Table 6-155 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-156 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Table 6-157 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the operation to perform. The value can be: create delete Enumeration values: • **create** • **delete**
tags	Yes	Array of **Tag** objects	Specifies the tags.

Table 6-158 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies any tag that is not included. Each tag can contain a maximum of 10 keys, and each key can contain a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. Resources not identified by different keys are in OR relationship, and values in one tag are in OR relationship. If no filtering condition is specified, full data is returned. Minimum: **1** Maximum: **128**
value	Yes	String	Specifies the request ID. Minimum: **0** Maximum: **255**

Response Parameters

None

Example Requests

- Batch adding tags to a transit IP address

```
POST https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/56121618-fb0a-4a51-aff0-e2eb9cba4c73/tags/action

{
  "action" : "create",
  "tags" : [ {
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value2"
  } ]
}
```

- Batch deleting tags from a transit IP address

```
POST https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/56121618-fb0a-4a51-aff0-e2eb9cba4c73/tags/action

{
  "action" : "delete",
  "tags" : [ {
```



```
    "key" : "key1",
    "value" : "value1"
  }, {
    "key" : "key2",
    "value" : "value2"
  } ]
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Tags added or deleted.

Error Codes

See **Error Codes**.

6.6.6 Deleting Tags from a Transit IP Address

Function

- This API is idempotent.
- When a tag is deleted, the tag character set is not verified. Before calling this API, encode the API URL. If the key of the tag to be deleted does not exist, 404 is displayed. The tag key cannot be left blank or be an empty string.

Debugging

You can debug this API through automatic authentication in or use the SDK sample code generated by API Explorer.

URI

DELETE /v3/{project_id}/transit-ips/{resource_id}/tags/{key}

Table 6-159 Path Parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. Minimum: **1** Maximum: **128**

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID. Minimum: **1** Maximum: **32**
resource_id	Yes	String	Specifies the ID of the transit IP address. Minimum: **36** Maximum: **36**

Request Parameters

Table 6-160 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the token of a user. It is a response to the API used to obtain the token of a user. This API is the only one that does not require authentication. The value of X-Subject-Token in the response header is the token. Minimum: **1** Maximum: **10240**

Response Parameters

None

Example Requests

This API is used to delete tags from a transit IP address.

```
DELETE https://{Endpoint}/v3/cfa563efb77d4b6d9960781d82530fd8/transit-ips/56121618-fb0a-4a51-aff0-e2eb9cba4c73/tags/key1
```

Example Responses

None

Status Codes

Status Code	Description
204	Tags deleted.

Error Codes

See **Error Codes**.

7 Permissions Policies and Supported Actions

7.1 Introduction

This section describes fine-grained permissions management for your NAT gateways. If your account does not need individual IAM users, then you may skip this section.

By default, new IAM users do not have permissions assigned. You need to add a user to one or more groups, and attach permissions policies or roles to these groups. Users inherit permissions from the groups to which they are added and can perform specified operations on cloud services based on the permissions.

You can grant users permissions by using **roles** and **policies**. Roles are a type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

Policy-based authorization is useful if you want to allow or deny the access to an API.

An account has all the permissions required to call all APIs, but IAM users must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users that have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user wants to query NAT gateways using an API, the user must have been granted permissions that allow the **nat:natGateways:list** action.

Supported Actions

NAT Gateway provides system-defined policies, which can be directly used in IAM. The account administrator can also create custom policies to supplement system-defined policies for more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Statements in a policy that allow or deny certain operations.
- APIs: REST APIs that can be called by a user who has been granted specific permissions.
- Actions: Specific operations that are allowed or denied.
- IAM or enterprise projects: Type of projects for which an action will take effect. Policies that contain actions for both IAM and enterprise projects can be used and take effect for both IAM and Enterprise Management. Policies that only contain actions for IAM projects can be used and only take effect for IAM. The account administrator can check whether an action supports IAM projects or enterprise projects in the action list. The check mark (√) indicates that the action supports the project and the cross symbol (×) indicates that the action does not support the project.

NAT Gateway supports the following actions that can be defined in custom policies:

- **NAT Gateway v2**, including actions supported by all v2 APIs of the NAT gateway, such as creating, updating, and deleting NAT gateways.
- **SNAT Rule v2**, including actions supported by all v2 APIs of the SNAT rule, such as creating and querying SNAT rules.
- **DNAT Rule v2**, including actions supported by all v2 APIs of the DNAT rule, such as creating and querying DNAT rules.

7.2 NAT Gateway v2

Permission	API	Action	IAM Project	Enterprise Project
Creating a NAT gateway	POST /v2/{project_id}/nat_gateways	nat:natGateways:create	√	√
Querying NAT gateways	GET /v2/{project_id}/nat_gateways	nat:natGateways:list	√	√
Querying details about a NAT gateway	GET /v2/{project_id}/nat_gateways/{nat_gateway_id}	nat:natGateways:get	√	√
Updating a NAT gateway	PUT /v2/{project_id}/nat_gateways/{nat_gateway_id}	nat:natGateways:update	√	√

Permission	API	Action	IAM Project	Enterprise Project
Deleting a NAT gateway	DELETE /v2/{project_id}/nat_gateways/{nat_gateway_id}	nat:natGateways:delete	√	√

7.3 SNAT Rule v2

Permission	API	Action	IAM Project	Enterprise Project
Creating an SNAT rule	POST /v2/{project_id}/snat_rules	nat:snatRules:create	√	√
Querying SNAT rules	GET /v2/{project_id}/snat_rules	nat:snatRules:list	√	√
Querying details about an SNAT rule	GET /v2/{project_id}/snat_rules/{snat_rule_id }	nat:snatRules:get	√	√
Deleting an SNAT rule	DELETE /v2/{project_id}/snat_rules/{snat_rule_id}	nat:snatRules:delete	√	√
Updating an SNAT rule	PUT /v2/{project_id}/snat_rules/{snat_rule_id}	nat:snatRules:update	√	√

7.4 DNAT Rule v2

Permission	API	Action	IAM Project	Enterprise Project
Creating a DNAT rule	POST /v2/{project_id}/dnat_rules	nat:dnatRules:create	√	√
Creating DNAT rules in batches	POST /v2/{project_id}/dnat_rules/batch	nat:dnatRules:create	√	√

Permission	API	Action	IAM Project	Enterprise Project
Querying DNAT rules	GET /v2/{project_id}/dnat_rules	nat:dnatRules:list	√	√
Querying details about a DNAT rule	GET /v2/{project_id}/dnat_rules/{dnat_rule_id }	nat:dnatRules:get	√	√
Deleting a DNAT rule	DELETE /v2/{project_id}/dnat_rules/{dnat_rule_id}	nat:dnatRules:delete	√	√
Updating a DNAT rule	PUT /v2/{project_id}/dnat_rules/{dnat_rule_id}	nat:dnatRules:update	√	√

7.5 NAT Gateway Tag v2.0

Permission	API	Action	IAM Project	Enterprise Project
Adding tags to a NAT gateway	POST /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags	nat:natGatewayTags:create	√	×
Querying NAT gateway tags	GET /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags	nat:natGatewayTags:get	√	×
Deleting NAT gateway tags	DELETE /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/{key}	nat:natGatewayTags:delete	√	×
Creating or deleting NAT Gateway tags in batches	POST /v2.0/{project_id}/nat_gateways/{nat_gateway_id}/tags/action	nat:natGatewayTags:create nat:natGatewayTags:delete	√	×

Permission	API	Action	IAM Project	Enterprise Project
Querying NAT Gateway resources by tag	POST /v2.0/{project_id}/nat_gateways/resource_instances/action	nat:natGatewayTags:get	√	×
Querying NAT gateway tags in a project	GET /v2.0/{project_id}/nat_gateways/tags	nat:natGatewayTags:get	√	×

8 Common Parameters

8.1 Status Codes

Normal Response Code	Type	Description
200	OK	Specifies the normal response code for the GET and PUT operations.
201	Created	Specifies the normal response code for the POST operation.
204	No Content	Specifies the normal response for the DELETE operation.

Error Response Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and password to access the requested page.
403 Forbidden	You are forbidden to access the requested page.
404 Not Found	The server could not find the requested page.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server cannot be accepted by the client.

Error Response Code	Description
407 Proxy Authentication Required	You must use the proxy server for authentication so that the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request cannot be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the server has received an invalid response.
503 Service Unavailable	Failed to complete the request because the system is currently unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2 Error Codes

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0001	Invalid value for created_at %(timestamp)s.	Invalid timestamp.	Make sure that you enter the time in the correct format.
400	NAT.0002	Invalid parameters.	Invalid request parameter.	Check whether the input parameter is correct.
400	NAT.0006	Rule has not been deleted.	NAT gateway deletion failed because its rule has not been deleted.	Check whether there are SNAT or DNAT rules using this NAT gateway.
400	NAT.0007	DB Error	The database is abnormal.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0008	Router %(router)s has no port for subnet %(subnet)s.	The subnet is not connected to the virtual router.	Add the subnet to the router port.
400	NAT.0009	Resource %(res_type)s %(res)s is used by %(user_type)s %(user)s	The resource is in use.	Check whether the resource is in use.
400	NAT.0010	Network %(network)s does not contain any IPv4 subnet	No subnets added in this VPC.	Check whether a subnet is added in this VPC. If no, add one.
400	NAT.0012	The network %(network)s already has nat gateway.	A NAT gateway has been created for this subnet.	Select a subnet for which no NAT gateway has been created.
400	NAT.0014	Invalid input for description. +exceeds maximum length of 255.	description contains more than 255 characters.	Enter a maximum of 255 characters.
400	NAT.0015	Invalid input for name. +exceeds maximum length of 255.	name contains more than 255 characters.	Enter a maximum of 255 characters.
400	NAT.0016	Invalid input for spec. Reason: '*' is not in ['1', '2', '3', '4'].	The value of spec is not 1, 2, 3, or 4.	Set spec to 1, 2, 3, or 4.
400	NAT.0017	Invalid input for router_id. Reason: '********-****-****-****-************' is not a valid UUID.	router_id is an invalid UUID.	Enter a valid router_id.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0018	Invalid input for internal_network_id. Reason: '********-****-****-****-************' is not a valid UUID.	internal_network_id is an invalid UUID.	Enter a valid internal_network_id.
400	NAT.0022	Either network_id or cidr must be specified. Both can not be specified at the same time	The subnet for which the SNAT rule is configured conflicts with the VPC subnet.	Specify either network_id or cidr.
400	NAT.0026	Floating IP ********-****-****-****-************ could not be found.	The ID of the floating IP address is not found.	Check whether the floating IP address ID is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0101	Lack of user authority. // request is null. // endpoint is empty. // resource type is invalid. // create natgateway requset is null. //update natgateway request is null // NatGateway id is invalid. //the enterprise project id is unsupported. //the enterprise project id in request is invalid. // request parameter is null. //tags is invalid. //get natgateways error limit is invalid. //get natgateways error marker is invalid. // Only admin user can do this action. // Parameters are invalid, check them and try.	NAT gateway request error.	Troubleshoot the fault as prompted or contact technical support.
400	NAT.0102	The system is busy. Please try again later.	The system is busy. Please try again later.	Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0103	NatGateway %(nat_gateway_id)s is not ACTIVE.	The NAT gateway is not activated.	If the NAT gateway is not in the running state for a long time, contact technical support.
400	NAT.0104	NatGateway %(nat_gateway_id)s is not UP. // NatGateway %(nat_gateway_id)s is frozen.can not update.	The NAT gateway is frozen.	Check the NAT gateway status. The gateway may be frozen due to arrears and cannot be updated.
400	NAT.0106	Concurrent conflict requests found	Concurrent operations on the NAT gateway conflict.	Contact technical support.
400	NAT.0107	Create NG Port failed.	Failed to create the internal port of the NAT gateway.	Contact technical support.
400	NAT.0108	NG Port %(port)s is unbound.	Failed to bind the internal port to the NAT gateway.	Contact technical support.
400	NAT.0109	NatGateway does not support IPv6.	NAT Gateway does not support IPv6 EIPs.	Bind an IPv4 EIP.
400	NAT.0110	Get Nat gateway host failed	Failed to select the gateway node.	Contact technical support.
400	NAT.0111	Get Nat gateway agent local_ip failed	Failed to obtain the gateway node IP address.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0112	Get RouteTable %(router_id)s failed.	Failed to obtain the Virtual Private Cloud (VPC) route table.	Contact technical support.
400	NAT.0113	%(limit)s NAT gateways has been created to this VPC, no more is allowed	The maximum number of NAT gateways has been reached.	Create a NAT gateway in another VPC or delete existing NAT gateways in this VPC.
400	NAT.0201	Endpoint is null or empty. // Endpoint is Invalid. // Request is null. // natGatewayId is invalid. // SnatRule id invalid. // NatGatewayId is invalid. // Invalid value for public ip id. //Endpoint is null. // request is null. //Query SnatRules list error marker is invalid.	Incorrect SNAT rule parameter.	Check whether the SNAT rule is correctly configured or contact technical support.
400	NAT.0202	Either network_id or cidr must be specified.Both can not be specified at the same time	The subnet for which the SNAT rule is configured conflicts with the VPC subnet.	Do not configure both Cidr and Network_id when you are configuring an SNAT rule.
400	NAT.0203	cidr is invalid, make sure it's format is correct.	Invalid CIDR block.	Enter a valid CIDR block, for example, 192.168.0.0/24.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0204	source_type and network_id is incompatible.	Invalid SNAT rule type.	If the SNAT rule is configured for servers in a VPC, Source_Type is optional or must be set to 0.
400	NAT.0205	cidr must be a subset of subnet's cidr.	The CIDR block is not a subset of the VPC subnet.	Enter a subset of the subnet CIDR block. For example, if the subnet is 192.168.0.0/24, cidr can be 192.168.0.0/25.
400	NAT.0206	cidr conflicts with subnet's cidr.	The CIDR block of the SNAT rule conflicts with the subnet CIDR block.	If the SNAT rule is configured for subnets connected to a VPC through Direct Connect or Cloud Connect, the CIDR block cannot conflict with the VPC subnet CIDR block.
400	NAT.0207	cidr in the request conflicts with cidrs of existing rules.	The CIDR block conflicts with the existing one.	Enter a CIDR block that does not conflict with existing ones.
400	NAT.0208	Snat rule for network %(network)s exists.	The rule already exists.	Select a subnet that has no SNAT rules configured.
400	NAT.0210	Invalid input for floating_ip_id. Reason: \'%(fip)s\' is not a valid UUID. // Invalid value for public ip id.	The public IP address UUID of the SNAT rule is invalid.	Enter a valid UUID.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0211	%(limit)s EIP has been associated to this SNAT rules's EIP pool, no more is allowed.	The maximum number of EIPs that can be bound to the SNAT rule has been reached.	Ensure that the number of EIPs does not exceed the maximum number allowed.
400	NAT.0212	SNAT Rule %(rule)s Associated or disassociate EIP %(fip)s Failed."	Failed to bind the EIP to or unbind the EIP from the SNAT rule.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0301	get dnatRules error limit is invalid. //get dnatrules error marker is invalid. // endpoint is empty. // DnatRule id invalid. //VPC ID is invalid. // Request is null. // DnatRule id invalid. // internal_service_port_range' out of range(1-65535). // internal_service_port_range': invalid format. // internal_service_port_range': param is null. //'internal_service_port_range' and 'external_service_port_range' must be equal. //for non-all port rule,the protocol can not be any. // param xxx is null in request body.	Incorrect DNAT rule parameter.	Check whether the DNAT rule is correctly configured or contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0302	Dnat rule protocol % (protocol)s not supported.Only protocol values % (values)s and integer representations [6, 17, 0] are supported.	Invalid DNAT rule protocol.	Configure a valid protocol. The number can be 6, 17, or 0, corresponding to protocols TCP, UDP, and ANY, respectively.
400	NAT.0303	Invalid value for port % (port)s	Invalid port in the DNAT rule.	Configure a valid internal port and external port. Supported range: 0 to 65535
400	NAT.0304	The port_id, private_ip, internal port and protocol specified have been occupied.	The internal network information in this DNAT rule conflicts with that in existing DNAT rules.	Enter a VM port ID, or private IP address and internal port, that does not conflict with existing DNAT rules.
400	NAT.0305	The floating ip, external port and protocol specified have been occupied.	The external network information in this DNAT rule conflicts with that in existing DNAT rules.	Enter a floating IP address ID, external port number, and protocol that do not conflict with those in existing DNAT rules.
400	NAT.0306	The external port equals 0 and internal port equals 0 and protocol equals any must satisfied at the same time.	Incorrect request for the DNAT rule.	Set both the internal port and external port to 0 and protocol to ANY to make the configurations take effect.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0307	The port_id already existing dnat allport rules or dnat_rules, can no longer create dnat rules or dnat allport rules.	The port ID in this DNAT rule conflicts with that in an existing DNAT rule.	Change the VM port ID to create a new DNAT rule or modify this DNAT rule.
400	NAT.0308	The private_ip already existing dnat allport rules or dnat rules, can no longer create dnat rules or dnat allport rules.	The private IP address configured in the DNAT rule conflicts with that in an existing DNAT rule.	Change the private IP address or modify this DNAT rule.
400	NAT.0309	%(limit)s DNAT rules has been associated to this NAT Gateway, no more is allowed	The maximum number of DNAT rules has been reached.	Ensure that the number of DNAT rules added to the NAT gateway is within the upper limit.
400	NAT.0310	The port_id and private_ip values are both empty, at least one value is not empty.	Some parameters of this DNAT rule are not configured.	Check whether port_id and private_ip are configured. If no, configure at least one of them.
400	NAT.0311	The private ip address is not legal.	Invalid private IP address in the DNAT rule.	Configure a valid private IP address.
400	NAT.0312	The virtual IP address is not supported.	Virtual IP addresses are not supported.	Configure a valid private IP address.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0313	%(limit)s DNAT rules has been associated to this Floating IP, no more is allowed	The maximum number of DNAT rules has been reached.	The maximum number of DNAT rules that are allowed to have the same floating IP address bound has been reached.
400	NAT.0314	batch create dnat rules max limit: %(limit)s	The maximum number of DNAT rules that can be added in batches has been reached.	Reduce the number of DNAT rules and then add them in batches.
400	NAT.0315	Port %(port)s is not a valid port.	Invalid VM port ID in the DNAT rule.	Configure a valid VM port ID.
400	NAT.0316	Vtep_ip is Null.	VtepIp must be specified.	Delete this DNAT rule and create a new one, or contact technical support.
400	NAT.0317	The port_id and private_ip exist at the same time and value is not empty, but at least one value is empty.	The DNAT rule contains mutually exclusive parameters.	Configure either port_id or private_ip.
400	NAT.0318	DNAT rule is frozen, can no longer update.	The DNAT rule has been frozen and cannot be updated.	Check whether the floating IP address bound to the DNAT rule is in arrears or whether the user account is in arrears.
400	NAT.0401	Floating Ip %(fip)s is freezed.	The EIP is frozen.	Select an EIP that is not frozen.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0402	Floating Ip %(fip)s has associated with port %(port)s.	The EIP has been bound to a port.	Select an EIP that has not been bound to any resource. For example, if an EIP has been bound to an ECS, it cannot be bound to a NAT gateway.
400	NAT.0403	There is a duplicate EIP %(fips)s in SNAT rule.	The EIP has been used by an SNAT rule.	Select another EIP.
400	NAT.0404	Floating Ip %(fip)s has used by nat gateway %(nat_gateway)s.	The EIP has been bound to a NAT gateway.	Select another EIP.
400	NAT.0405	Floating Ip %(fip)s has been occupied.	The EIP is in use.	Select another EIP.
400	NAT.0407	Floating Ip %(fip)s is used by other rules	The EIP has been bound to a rule.	Select an EIP that is not in use.
400	NAT.0408	Floating Ip %(fip)s can not be associated with both DNAT rule and DNAT all port rule.	A DNAT rule cannot share an EIP with another DNAT rule in which mapping to a specific port is not set.	Select another EIP.
400	NAT.0409	Floating Ip %(fip)s can not be associated with both SNAT rule and DNAT all port rule.	An SNAT rule cannot share an EIP with a DNAT rule in which mapping to a specific port is not set.	Select another EIP.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.0410	Invalid value of the FloatIP.	Invalid floating IP address.	Enter a valid floating IP address.
400	VPC.0002	Available zone Name is null.	The AZ is left blank.	Check whether availability_zone in the request body for creating a subnet is left blank.
400	VPC.0004	VPC does not active,please try later.	The VPC status is abnormal.	Try again later or contact technical support.
400	VPC.0007	urlTenantId is not equal tokenTenantId	The tenant ID in the URL is different from that parsed in the token.	Contact technical support.
400	VPC.0011	EnterpriseProjectId is invalid	Invalid enterprise project ID.	Enter a valid enterprise project ID.
400	VPC.0014	This enterpriseProject status is disable.	Unavailable enterprise project.	Use the ID of an available enterprise project.
400	VPC.2000	Lack of user authority. // request is null. // endpoint is empty. // resource type is invalid. // create natgateway requset is null. //update natgateway request is null.	NAT gateway request error.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2001	NatGateway id is invalid. //the enterprise project id in request is invalid. // request parameter is null. //tags is invalid. //get natgateways error limit is invalid. //get natgateways error marker is invalid. // Only admin user can do this action. // Parameters are invalid, check them and try.	Incorrect NAT gateway parameter.	Check whether the NAT gateway is correctly configured or contact technical support.
400	VPC.2002	Invalid parameters.	Invalid request parameter.	Check whether the input parameter is correct.
400	VPC.2004	NatGateway %(nat_gateway_id)s is not ACTIVE.	The NAT gateway is not activated.	If the NAT gateway is not in the running state for a long time, contact technical support.
400	VPC.2005	NatGateway %(nat_gateway_id)s is not UP.	The NAT gateway is not in the UP state.	Check whether the gateway has been frozen due to arrears or other reasons.
400	VPC.2006	NatGateway %(nat_gateway_id)s is frozen.can not update	The NAT gateway is frozen.	Check whether the gateway is frozen due to arrears. If yes, it cannot be updated.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2007	NatGateway %(nat_gateway_id)s does not exist.	The NAT gateway is not found.	Check whether the NAT gateway ID is correct.
400	VPC.2008	Network %(network)s does not contain any IPv4 subnet	No subnets added in this VPC.	Contact technical support.
400	VPC.2009	Network %(network_id)s does not exist.	The subnet is not found.	Enter a valid subnet.
400	VPC.2010	The router %(router_id)s has default route.	The default route already exists.	Delete the default route and then create a NAT gateway.
400	VPC.2011	The router %(router_id)s does not exist.	The router is not found.	Check whether the entered router ID is correct.
400	VPC.2012	The router %(router_id)s already has nat gateway.	The VPC already has a NAT gateway.	Select another VPC.
400	VPC.2013	Router %(router)s has no port for subnet %(subnet)s.	The subnet is not connected to the virtual router.	Add the subnet to the router port.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2014	Endpoint is null or empty. // Endpoint is Invalid. // Request is null. // natGatewayId is invalid. // SnatRule id invalid. // NatGatewayId is invalid. // Invalid value for public ip id. //Endpoint is null. // request is null. //Query SnatRules list error marker is invalid.	Incorrect SNAT rule parameter.	Check whether the SNAT rule is correctly configured or contact technical support.
400	VPC.2016	Rule has not been deleted.	NAT gateway deletion failed because its rule has not been deleted.	Contact technical support.
400	VPC.2018	Snat rule for network %(network)s exists.	The rule already exists.	Select a subnet that has no SNAT rules configured.
400	VPC.2019	Resource %(res_type)s %(res)s is used by %(user_type)s %(user)s	The resource is in use.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2020	get dnatRules error limit is invalid. //get dnatrules error marker is invalid. // endpoint is empty. // DnatRule id invalid. // Request is null. // DnatRule id invalid. // DnatRule natGatewayId id invalid.	Incorrect DNAT rule parameter.	Check whether the DNAT rule is correctly configured or contact technical support.
400	VPC.2022	Port %(port)s is not a valid port.	Invalid VM port ID in the DNAT rule.	Configure a valid VM port ID.
400	VPC.2023	The port_id, private_ip, internal port and protocol specified have been occupied.	The internal network information in this DNAT rule conflicts with that in existing DNAT rules.	Enter a VM port ID, or private IP address and internal port, that does not conflict with existing DNAT rules.
400	VPC.2024	The floating ip, external port and protocol specified have been occupied.	The external network information in this DNAT rule conflicts with that in existing DNAT rules.	Enter a floating IP address ID, external port number, and protocol that do not conflict with those in existing DNAT rules.
400	VPC.2026	%(limit)s DNAT rules has been associated to this Floating IP, no more is allowed	The maximum number of DNAT rules that are allowed to have the same floating IP address bound has been reached.	Reduce the number of DNAT rules.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2027	The port_id already existing dnat allport rules or dnat_rules, can no longer create dnat rules or dnat allport rules.	The port ID in this DNAT rule conflicts with that in an existing DNAT rule.	Change the VM port ID to create a new DNAT rule or modify this DNAT rule.
400	VPC.2028	The private_ip already existing dnat allport rules or dnat rules, can no longer create dnat rules or dnat allport rules.	The private IP address configured in the DNAT rule conflicts with that in an existing DNAT rule.	Change the private IP address or modify this DNAT rule.
400	VPC.2029	DNAT rule is frozen, can no longer update.	The DNAT rule has been frozen and cannot be updated.	Check whether the floating IP address bound to the DNAT rule is in arrears or whether the user account is in arrears.
400	VPC.2030	The system is busy. Please try again later.	The system is busy. Please try again later.	Try again later.
400	VPC.2031	Either network_id or cidr must be specified.Both can not be specified at the same time	The subnet for which the SNAT rule is configured conflicts with the VPC subnet.	Do not configure both Cidr and Network_id when you are configuring an SNAT rule.
400	VPC.2032	cidr is invalid, make sure it's format is correct.	Invalid CIDR block.	Enter a valid CIDR block, for example, 192.168.0.0/24.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2033	source_type and network_id is incompatible.	Invalid SNAT rule type.	If the SNAT rule is configured for servers in a VPC, Source_Type is optional or must be set to 0. If the SNAT rule is configured for servers in your on-premises data centers that are connected to a VPC through Direct Connect or your servers in another VPC, Source_Type must be set to 1.
400	VPC.2034	cidr must be a subset of subnet's cidr.	The CIDR block is not a subset of the VPC subnet.	Enter a subset of the subnet CIDR block. For example, if the subnet is 192.168.0.0/24, cidr can be 192.168.0.0/25.
400	VPC.2035	cidr conflicts with subnet's cidr.	The CIDR block of the SNAT rule conflicts with the subnet CIDR block.	If the SNAT rule is configured for subnets connected to a VPC through Direct Connect or Cloud Connect, the CIDR block cannot conflict with the VPC subnet CIDR block.
400	VPC.2036	cidr in the request conflicts with cidrs of existing rules.	The CIDR block conflicts with the existing one.	Enter a CIDR block that does not conflict with existing ones.
400	VPC.2037	The virtual IP address is not supported.	Virtual IP addresses are not supported.	Configure a valid private IP address.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2038	%(limit)s DNAT rules has been associated to this NAT Gateway, no more is allowed	The maximum number of DNAT rules has been reached.	Delete some DNAT rules.
400	VPC.2039	%(limit)s EIP has been associated to this SNAT rules's EIP pool, no more is allowed.	The maximum number of EIPs bound to the SNAT rule has been reached.	Reduce the number of EIPs.
400	VPC.2040	Invalid value for public ip id.	The public IP address ID of an SNAT rule cannot be left blank.	Enter a valid UUID.
400	VPC.2042	There is a duplicate EIP %(fips)s in SNAT rule.	The EIP has been used by an SNAT rule.	Select another EIP.
400	VPC.2043	Floating Ip %(fip)s is used by other rules	The EIP has been bound to a rule.	Select another EIP.
400	VPC.2044	Invalid input for floating_ip_id. Reason: \'%(fip)s\' is not a valid UUID.	The public IP address UUID of the SNAT rule is invalid.	Enter a valid UUID.
400	VPC.2045	Get Nat gateway host failed	Failed to select the gateway node.	Contact technical support.
400	VPC.2046	Get Nat gateway agent local_ip failed	Failed to obtain the gateway node IP address.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2047	Get RouteTable %(router_id)s failed.	Failed to obtain the VPC route table.	Contact technical support.
400	VPC.2048	Invalid value for created_at %(timestamp)s.	Invalid timestamp.	Make sure that you enter the time in the correct format.
400	VPC.2049	DB Error	The database is abnormal.	Contact technical support.
400	VPC.2050	Concurrent conflict requests found	Concurrent operations on the NAT gateway conflict.	Contact technical support.
400	VPC.2051	Create NG Port failed.	Failed to create the internal port of the NAT gateway.	Contact technical support.
400	VPC.2052	NG Port %(port)s is unbound.	Failed to bind the internal port to the NAT gateway.	Contact technical support.
400	VPC.2053	NatGateway does not support IPv6.	NAT Gateway does not support IPv6 EIPs.	Bind an IPv4 EIP.
400	VPC.2054	Dnat rule protocol %(protocol)s not supported.Only protocol values %(values)s and integer representations [6, 17, 0] are supported.	Invalid DNAT rule protocol.	Configure a valid protocol. The number can be 6, 17, or 0, corresponding to protocols TCP, UDP, and ANY, respectively.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2055	The port_id and private_ip exist at the same time and value is not empty, but at least one value is empty.	The DNAT rule contains mutually exclusive parameters.	Configure either port_id or private_ip.
400	VPC.2056	The port_id and private_ip values are both empty, at least one value is not empty.	Some parameters of this DNAT rule are not configured.	Check whether port_id and private_ip are configured. If no, configure at least one of them.
400	VPC.2057	batch create dnat rules max limit: % (limit)s	The maximum number of DNAT rules allowed to be added in batches has been exceeded.	Reduce the number of DNAT rules for batch addition.
400	VPC.2058	Vtep_ip is Null.	VtepIp must be specified.	Contact technical support.
400	VPC.2059	Floating Ip % (fip)s is freezed.	The EIP is frozen.	Select an EIP that is not frozen.
400	VPC.2060	Floating Ip % (fip)s has associated with port % (port)s.	The EIP has been bound to a port.	Select an EIP that has not been bound to any resource. For example, if an EIP has been bound to an ECS, it cannot be bound to a NAT gateway.
400	VPC.2061	Floating Ip % (fip)s has used by nat gateway % (nat_gateway)s.	The EIP has been bound to a NAT gateway.	Select another EIP.

Status Code	Error Codes	Error Message	Description	Solution
400	VPC.2062	Floating Ip % (fip)s has been occupied.	The EIP is in use.	Select another EIP.
400	VPC.2069	Invalid value for port % (port)s	Invalid port in the DNAT rule.	Configure a valid internal port and external port. Supported range: 0 to 65535
400	VPC.2070	The external port equals 0 and internal port equals 0 and protocol equals any must satisfied at the same time.	Incorrect request for the DNAT rule.	Set the private port number and public port number to 0 and Protocol to Any.
400	VPC.2071	The private ip address is not legal.	Invalid private IP address in the DNAT rule.	Configure a valid private IP address.
400	VPC.2073	Floating Ip % (fip)s can not be associated with both DNAT rule and DNAT all port rule.	A DNAT rule cannot share an EIP with another DNAT rule in which mapping to a specific port is not set.	Select another EIP.
400	VPC.2074	Floating Ip % (fip)s can not be associated with both SNAT rule and DNAT all port rule.	An SNAT rule cannot share an EIP with another DNAT rule in which mapping to a specific port is not set.	Select another EIP.
400	VPC.2075	Enter a maximum of 255 characters.	The description contains more than 255 characters.	Enter a maximum of 255 characters.

Status Code	Error Codes	Error Message	Description	Solution
400/404	NAT.0105	NatGateway % (nat_gateway _id)s does not exist.	The NAT gateway is not found. (HTTP status code 400 indicates that the gateway to be deleted is not found. HTTP status code 404 indicates that the gateway that you created or queried is not found.)	Check whether the NAT gateway ID is available.
401	NAT.0025	Token is expired.	Token expired.	Check whether the token is within the validity period.
401	VPC.0008	Invalid token in the header.	Invalid token.	Check whether the token in the request header is valid.
401	VPC.0009	real-name authenticatio n fail.	Real-name authenticatio n failed.	Contact technical support.
403	VPC.0010	Rules on xx by ** disallowed by policy	Insufficient permissions to call the underlying system.	Obtain required permissions.
403	VPC.2201	Policy doesn't allow <x:x:x> to be performed	Insufficient fine-grained permissions.	Obtain required permissions.

Status Code	Error Codes	Error Message	Description	Solution
403	VPC.2701	Token not allowed to do this action.	You do not have permissions to perform this operation, or your account balance is insufficient.	Check whether your account balance is insufficient or whether your account has been frozen.
404	NAT.0004	The router %(router_id)s does not exist.	The router is not found.	Check whether the entered router ID is correct.
404	NAT.0005	Network %(network_id)s does not exist.	The subnet is not found.	Enter a valid subnet.
404	NAT.0013	Router %(router)s for the specified NAT gateway could not be found.	The route for the specified NAT gateway is not found.	Create a route for the specified NAT gateway.
404	NAT.0019	Network ********-****-****-****-************ could not be found.	The subnet ID is not found.	Check whether the subnet ID is available.
404	NAT.0020	Specifying 'tenant_id' other than authenticated tenant in request requires admin privileges	tenant_id is left blank or not found.	Check whether the tenant ID is available.

Status Code	Error Codes	Error Message	Description	Solution
404	NAT.0021	Invalid input for nat_gateway_id. Reason: '********-****-****-****-************'is not a valid UUID.	Nat_gateway_id is left blank or not found.	Check whether the NAT gateway ID is available.
404	NAT.0023	Port '********-****-****-****-************' could not be found.	The port ID is not found.	Check whether the port ID is available.
404	NAT.0024	Invalid input for floating_ip_id. Reason: '********-****-****-****-************'is not a valid UUID.	Floating_ip_id is left blank, not found, or is invalid.	Check whether the floating IP address ID is correct.
404	NAT.0209	No Snat Rule exist with id %(id)s	The SNAT rule is not found.	Check whether the SNAT rule ID is available.
404	NAT.0319	No Dnat Rule exist with id %(id)s	The DNAT rule is not found.	Contact technical support.
404	VPC.0003	VPC does not exit.	The VPC is not found.	Check whether the VPC ID is valid or whether the VPC is available.
400	NAT.1010	Request parameter Json parsing failed %s.	Parsing JSON request failed.	Check whether the request parameters are in the correct format.
400	NAT.1015	Tags parameter is illegal.	Invalid tag.	Enter a valid tag value.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.1016	The number of tags exceeds the limit %s.	The maximum number of tags has been reached.	Add up to %s tags.
400	NAT.1022	Account is restricted, operation is forbidden.	The private NAT gateway cannot be created because your account has been suspended.	Check whether your account is suspended and contact technical support.
400	NAT.1023	Account is suspended, operation is forbidden.	The private NAT gateway cannot be created because your account has been frozen.	Check whether your account is frozen and contact technical support.
400	NAT.1101	Transit subnet can not be create with this network %s. //External subnet can not be create with this network %s.	The subnet where the transit IP address is located is unavailable.	Contact technical support.
400	NAT.1201	%s downlink vpc has been associated to this private nat gateway, no more is allowed.	Only one downlink_vpc can be entered.	Enter only one downlink_vpc.
400	NAT.1202	There are one or more rules still in use on the gateway %s, can not be deleted.	The private NAT gateway cannot be deleted because it has rules.	Delete a private NAT gateway after deleting all its rules.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.1204	This vpc at most have %s private nat gateway.	The maximum number of private NAT gateways in the VPC has been reached.	Use another VPC or delete other private NAT gateways in this VPC.
400	NAT.1206	Update (%s) spec to (%s) are the same as the original.	The private NAT gateway can be updated only to different specifications.	Update the private NAT gateway to different specifications.
400	NAT.1207	Gateway %s already has %d rules, the maximum number of rule for spec %s is %d. Downgrade is forbidden. //Add this gateway rule over rule max %d.	The maximum number of rules that can be added on the private NAT gateway of the new specifications has been exceeded, so the gateway specifications cannot be updated. The maximum number of NAT gateway rules has been reached. No more rules can be created.	Ensure that the number of rules added on the private NAT gateway does not exceed the maximum number of rules after the update. Delete some rules or update the private NAT gateway to more robust specifications.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.1208	Gateway %s is frozen, create operation is forbidden. // Gateway %s is frozen, delete operation is forbidden. // Gateway %s is frozen, update operation is forbidden.	Operation not allowed. The private NAT gateway has been frozen.	Check whether your private NAT gateway is frozen and contact technical support.
400	NAT.1306	dnat Parameters entered are illegal, protocol:any should be entered with internal_service_port:0 and transit_service_port:0 together. // dnat Parameters entered are illegal, protocol:any should be entered with internal_service_port:0 and external_service_port:0 together.	Invalid parameters in the request body of the DNAT rule.	Set protocol to any, internal_service_port to 0, and transit_service_port to 0.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.1311	The networkInterfaceId(port_id) and private_ip_address(fixed_ip_address) exist at the same time or both are empty, but at least one value is empty.	Either network_interface_id or private_ip_address must be specified.	Specify network_interface_id or private_ip_address.
400	NAT.1404	No more IP addresses available on subnet %s.	There are no available IP addresses in the subnet.	Contact technical support.
400	NAT.1405	IP address %s is not a valid IP for the subnet.	Invalid IP address in the subnet.	Contact technical support.
400	NAT.1501	Either virSubnet(network) or cidr must be specified." + "Both can not be specified at the same time.	Either virsubnet_id or cidr must be specified.	Specify virsubnet_id or cidr.
400	NAT.1502	Cidr is invalid, make sure it's format is correct.	Invalid cidr.	Enter a valid CIDR block, for example, 192.168.0.0/24.

Status Code	Error Codes	Error Message	Description	Solution
400	NAT.1506	%s transit ip has been associated to this SNAT rules's transit ip pool, no more is allowed. //%s external ip has been associated to this SNAT rules's external ip pool, no more is allowed.	The maximum number of transit IP addresses that can be selected for the SNAT rule has been exceeded.	Enter only one transit IP address.
400	NAT.1507	'transit_ip_id' attribute value should be 'uuid' type! //'external_ip_id' attribute value should be 'uuid' type!	Invalid transit_ip_id.	Enter the transit_ip_id value in UUID format.
404	NAT.1002	%s %s could not be found.	The resource is not found.	Check whether the resource is available or contact technical support.
404	NAT.1003	VirSubnet %s could not be found in vpc for gateway %s.	The subnet for which the SNAT rule is configured is not in the VPC of the private NAT gateway.	Ensure that the subnet for which the SNAT rule is configured is in the VPC of the private NAT gateway.
404	NAT.1009	Port %s information is missing.	Port information is missing.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
409	NAT.1303	Transit ip %s is in this vpc, not support to specified. // External ip %s is in this vpc, not support to specified.	The VPC of the transit IP address must be different from that of the private NAT gateway.	Use a transit IP address that is in a different VPC of the private NAT gateway.
409	NAT.1304	Port %s is not in this vpc, not support to specified.	The port for which the DNAT rule is configured is not in the VPC of the private NAT gateway.	Ensure that the port for which the DNAT rule is configured is in the VPC of the private NAT gateway.
409	NAT.1305	Transit ip %s is used by exist dnat rule. // External ip %s is used by exist dnat rule.	The transit IP address is already used by a DNAT rule.	Use a transit IP address that is not used by any DNAT rule.
409	NAT.1307	Internal parameters entered conflict with exist dnat rules. // PrivateIp(FixedIp) %s is used by exist dnat rule.	The internal network information in this DNAT rule conflicts with that in existing DNAT rules.	Use a port, private IP address, backend port, and protocol that do not conflict with those of existing DNAT rules.
409	NAT.1308	Transit ip %s is used by exist snat rule. // External ip %s is used by exist snat rule.	The transit IP address is already used by an SNAT rule.	Use a transit IP address that is not used by any SNAT rule.

Status Code	Error Codes	Error Message	Description	Solution
409	NAT.1309	Port %s is used by exist dnat rule.	The port is being used by a DNAT rule of another protocol type.	Use a port that is not used by DNAT rules of another protocol type.
409	NAT.1310	Transit parameters entered conflict with exist dnat rules. // External parameters entered conflict with exist dnat rules.	The external network information in this DNAT rule conflicts with the existing one.	Use a transit IP address, port number, and protocol that do not conflict with those in existing DNAT rules.
409	NAT.1403	Unable to complete operation for subnet %s. The IP address %s is in use.	This IP address is already in use.	Contact technical support.
409	NAT.1406	Transit ip %s has used by private nat gateway %s. //External ip %s has used by private nat gateway %s.	The transit IP address has been used by another private NAT gateway.	Use a transit IP address that is not used by other private NAT gateways.
409	NAT.1409	Transit ip %s is used by rules. // External ip %s is used by rules.	The transit IP address is used by a rule.	Ensure that the transit IP address is not used by other rules before deleting it.

Status Code	Error Codes	Error Message	Description	Solution
409	NAT.1410	Transit ip %s is used by dnat rules of other protocols. // External ip %s is used by dnat rules of other protocols.	The transit IP address is being used by a DNAT rule of another protocol type.	Use a transit IP address that is not used by DNAT rules of another protocol type.
409	NAT.1503	Snat rule for network %s exists.	An SNAT rule has been configured for this subnet.	Select a subnet that has no SNAT rules configured.
409	NAT.1505	Snat rule for cidr %s exists.	An SNAT rule has been configured for this CIDR block.	Enter a CIDR block that does not conflict with existing ones.
500	NAT.1001	Internal Server Error.	Internal service error.	Contact technical support.
500	NAT.1004	Create Port Failed with subnet %s.	Failed to create the port in the subnet.	Contact technical support.
500	NAT.1005	Delete Port %s Failed.	Port deletion failed.	Contact technical support.

8.3 Obtaining a Project ID

Scenarios

A project ID is required for some URLs when an API is called. Therefore, you need to obtain a project ID in advance. Two methods are available:

- **Obtain the Project ID by Calling an API**
- **Obtain the Project ID from the Console**

Obtain the Project ID by Calling an API

You can obtain a project ID by calling the API used to **query projects based on specified criteria**.

The API used to obtain a project ID is GET https://{Endpoint}/v3/projects. {Endpoint} is the IAM endpoint and can be obtained from Regions and Endpoints. For details about API authentication, see **Authentication**.

The following is an example response. The value of **id** is the project ID.

```
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "project_name",
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```

Obtain a Project ID from the Console

To obtain a project ID from the console, perform the following operations:

1. Log in to the management console.
2. Click the username and select **My Credentials** from the drop-down list. On the **API Credentials** page, view the project ID in the project list.

8.4 Resource Status Description

Table 8-1 Resource status description

Status	Description
ACTIVE	The resource status is normal.
PENDING_CREATE	The resource is being created.
PENDING_UPDATE	The resource is being updated.
PENDING_DELETE	The resource is being deleted.
EIP_FREEZED	The EIP of the resource is frozen.
INACTIVE	The resource status is abnormal.

A Change History

Released On	What's New
2023-04-03	This issue is the second official release, which incorporates the following changes: • Added **Private NAT Gateways** in **API Overview**. • Added **Private Nat API**. • Updated **Error Codes**.
2022-08-30	This issue is the first official release.

