

Elastic Cloud Server

API Reference

Issue 01
Date 2025-01-07



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property 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 Cloud 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
2 API Overview	3
3 Selecting an API Type or Version	6
4 Calling APIs	12
4.1 Making an API Request	12
4.2 Authentication	16
4.3 Response	17
5 APIs (Recommended)	20
5.1 Lifecycle Management	20
5.1.1 Creating an ECS	20
5.1.2 Deleting ECSs	32
5.2 Status Management	35
5.2.1 Reinstalling an ECS OS (Using an Image with Cloud-Init Installed)	35
5.2.2 Changing an ECS OS (Using an Image with Cloud-Init Installed)	38
5.2.3 Cold Migrating an ECS	41
5.2.4 Modifying the Specifications of an ECS	43
5.3 Batch Operations	45
5.3.1 Starting ECSs in a Batch	45
5.3.2 Restarting ECSs in a Batch	47
5.3.3 Stopping ECSs in a Batch	48
5.4 Flavor Management	50
5.4.1 Querying Details About Flavors and Extended Flavor Information	50
5.5 NIC Management	58
5.5.1 Adding NICs to an ECS in a Batch	58
5.5.2 Deleting NICs from an ECS in a Batch	60
5.6 Disk Management	61
5.6.1 Attaching a Disk to an ECS	62
5.6.2 Detaching an EVS Disk from an ECS	64
5.6.3 Querying Disk Attachment of an ECS	66
5.6.4 Querying a Single Disk Attached to an ECS	67
5.7 Tenant Quota Management	69
5.7.1 Querying Tenant Quotas	69

5.8 Task Status Management.....	72
5.8.1 Querying Task Execution Status.....	72
5.9 Tag Management.....	75
5.9.1 Tag Types.....	76
5.9.2 Querying ECSs by Tag.....	76
5.9.3 Adding Tags to an ECS in a Batch.....	81
5.9.4 Deleting Tags from an ECS in a Batch.....	83
5.9.5 Querying Project Tags.....	85
5.9.6 Querying Tags of an ECS.....	87
5.10 ECS Group Management.....	88
5.10.1 Creating an ECS Group.....	88
5.10.2 Deleting an ECS Group.....	90
5.10.3 Adding an ECS to an ECS Group.....	92
5.10.4 Removing an ECS from an ECS Group.....	93
5.10.5 Querying ECS Groups.....	95
5.10.6 Querying Details About an ECS Group.....	97
6 Native OpenStack Nova APIs.....	99
6.1 API Version Query.....	99
6.1.1 Querying All API Versions.....	99
6.1.2 Querying a Specified API Version.....	103
6.2 Lifecycle Management.....	107
6.2.1 Creating an ECS.....	107
6.2.2 Modifying an ECS.....	122
6.2.3 Deleting an ECS.....	127
6.2.4 Querying the ECS List.....	128
6.2.5 Querying Details About ECSs.....	131
6.2.6 Querying Details About an ECS.....	141
6.3 Status Management.....	149
6.3.1 Starting an ECS.....	150
6.3.2 Restarting an ECS.....	151
6.3.3 Disabling an ECS.....	152
6.3.4 Locking an ECS.....	153
6.3.5 Unlocking an ECS.....	155
6.3.6 Rebuilding an ECS.....	156
6.3.7 Creating an Image Using an ECS.....	160
6.3.8 Modifying the Specifications of an ECS.....	163
6.3.9 Confirming the Specifications Modification of an ECS.....	164
6.3.10 Rolling Back ECS Specifications Modification.....	165
6.3.11 Adding an ECS to the Monitoring List.....	167
6.4 Network Management.....	168
6.4.1 Querying Networks.....	168
6.4.2 Querying the Networks of a Specified ECS.....	170

6.4.3 Querying the Specified Network of an ECS.....	172
6.5 Security Group Management.....	173
6.5.1 Adding a Security Group.....	173
6.5.2 Removing a Security Group.....	175
6.5.3 Querying Security Groups of a Specified ECS.....	176
6.6 Flavor Management.....	180
6.6.1 Querying ECS Flavors.....	180
6.6.2 Querying Details About ECS Flavors.....	183
6.6.3 Querying Details About an ECS Flavor.....	186
6.6.4 Querying the extra_specs Value for an ECS Flavor.....	189
6.7 NIC Management.....	190
6.7.1 Querying NICs of an ECS.....	190
6.7.2 Querying Details About a Specified NIC of an ECS.....	192
6.7.3 Adding a NIC to an ECS.....	194
6.7.4 Deleting a NIC from an ECS.....	197
6.8 Disk Management.....	198
6.8.1 Querying Disks Attached to an ECS.....	198
6.8.2 Querying a Disk Attached to an ECS.....	199
6.8.3 Attaching an ECS Data Disk.....	201
6.8.4 Detaching a Disk from an ECS.....	203
6.9 Metadata Management.....	206
6.9.1 Updating ECS Metadata.....	206
6.9.2 Configuring ECS Metadata.....	208
6.9.3 Deleting Specified ECS Metadata.....	210
6.9.4 Querying ECS Metadata.....	211
6.9.5 Obtaining ECS Metadata with a Specified Key.....	212
6.9.6 Modifying ECS Metadata with a Specified Key.....	213
6.10 Tenant Quota Management.....	215
6.10.1 Querying Tenant Quota Limits.....	215
6.10.2 Querying Tenant Quotas.....	218
6.10.3 Querying Default Quotas.....	221
6.11 Key and Password Management.....	223
6.11.1 Querying SSH Key Pairs.....	223
6.11.2 Querying a Specified SSH Key Pair.....	225
6.11.3 Creating and Importing an SSH Key Pair.....	227
6.11.4 Deleting an SSH Key.....	230
6.11.5 Obtaining the Password of an ECS.....	231
6.11.6 Deleting the Password of an ECS.....	232
6.12 ECS Group Management.....	233
6.12.1 Creating an ECS Group.....	233
6.12.2 Querying ECS Groups.....	236
6.12.3 Querying Details About an ECS Group.....	237

6.12.4 Deleting an ECS Group.....	239
6.13 ECS Operation Management.....	240
6.13.1 Querying Operations on an ECS.....	240
6.13.2 Querying ECS Operations by Request ID.....	242
6.14 ECS Console Management.....	244
6.14.1 Obtaining ECS Management Console Logs.....	244
6.15 Tag Management.....	246
6.15.1 Querying All Tags of an ECS.....	246
6.15.2 Adding Tags to an ECS.....	247
6.15.3 Deleting All Tags of an ECS.....	248
6.15.4 Adding a Tag to an ECS.....	249
6.15.5 Querying a Specified Tag for an ECS.....	251
6.15.6 Deleting a Specified Tag from an ECS.....	252
6.16 Historical Versions.....	253
7 Application Examples.....	254
7.1 Creating an ECS.....	254
8 Data Structure.....	261
8.1 Data Structure for Creating ECSs.....	261
8.2 Data Structure for Querying Details About ECSs.....	270
8.3 Data Structure for Querying Details About Specifications.....	273
9 Permissions and Supported Actions.....	279
9.1 Introduction.....	279
9.2 Actions Supported by Policy-based Authorization.....	280
9.2.1 Lifecycle Management.....	281
9.2.2 ECS Status Management.....	284
9.2.3 Batch Operations.....	286
9.2.4 Network Management.....	286
9.2.5 Image Management.....	286
9.2.6 Security Group Management.....	287
9.2.7 Specifications Query.....	289
9.2.8 NIC Management.....	290
9.2.9 Disk Management.....	293
9.2.10 Metadata Management.....	295
9.2.11 Tenant Quota Management.....	296
9.2.12 SSH Key Management.....	296
9.2.13 Password Management.....	297
9.2.14 Floating IP Address Management.....	298
9.2.15 ECS Group Management.....	299
9.2.16 Tag Management.....	300
10 Common Parameters.....	303
10.1 Returned Values for General Requests.....	303

10.2 Obtaining a Project ID.....	304
10.3 Task Request Result.....	305
10.3.1 Responses (Task).....	305
10.3.2 Returned Values.....	306
10.4 Batch Task Request.....	307
10.4.1 Responses (Batch Operation).....	307
11 Out-of-Date APIs.....	310
11.1 Status Management.....	310
11.1.1 Querying Automatic Recovery of an ECS (Discarded).....	310
11.1.2 Managing Automatic Recovery of an ECS (Discarded).....	311
11.2 Image Management (OpenStack Nova APIs).....	312
11.2.1 Querying Images (Discarded).....	312
11.2.2 Querying Image Details (Discarded).....	316
11.2.3 Querying Details About a Specified Image (Discarded).....	320
11.2.4 Querying the Metadata of a Specified Image (Discarded).....	323
11.2.5 Deleting an Image (Discarded).....	325
11.3 Security Group Management (OpenStack Nova APIs).....	325
11.3.1 Querying Security Groups (Discarded).....	326
11.3.2 Creating a Security Group (Discarded).....	330
11.3.3 Querying Details About a Security Group (Discarded).....	332
11.3.4 Updating a Security Group (Discarded).....	335
11.3.5 Deleting a Security Group (Discarded).....	339
11.3.6 Creating a Security Group Rule (Discarded).....	340
11.3.7 Deleting a Security Group Rule (Discarded).....	344
11.4 Disk Management (OpenStack Nova APIs).....	345
11.4.1 Querying Brief Information About Disks (Discarded).....	345
11.4.2 Querying Detailed Information About Disks (Discarded).....	347
11.4.3 Querying Information About a Disk (Discarded).....	349
11.4.4 Creating a Disk (Discarded).....	351
11.4.5 Deleting a Disk (Discarded).....	354
11.5 Floating IP Address Management (OpenStack Nova APIs).....	355
11.5.1 Binding a Floating IP Address (Discarded).....	355
11.5.2 Unbinding a Floating IP Address (Discarded).....	356
11.5.3 Assigning a Floating IP Address (Discarded).....	358
11.5.4 Querying Floating IP Addresses (Discarded).....	359
11.5.5 Querying Details About a Floating IP Address (Discarded).....	361
11.5.6 Releasing a Floating IP Address (Discarded).....	363
11.5.7 Querying Floating IP Address Pools (Discarded).....	364
11.6 Snapshot Management (OpenStack Nova APIs).....	365
11.6.1 Creating a Snapshot (Discarded).....	365
11.6.2 Querying Snapshots (Discarded).....	368
11.6.3 Deleting a Snapshot (Discarded).....	369

A Appendix.....	371
A.1 HTTP Status Codes.....	371
A.2 Error Codes.....	372
A.3 ECS Statuses.....	390

1 Before You Start

Overview

Welcome to *Elastic Cloud Server API Reference*. An Elastic Cloud Server (ECS) is an easy-to-obtain, elastically scalable compute server that consists of a CPU, memory, image, and EVS disks. An ECS can work with a Virtual Private Cloud (VPC), virtual firewall, and multiple copies of data to build an efficient, reliable, and secure computing environment to let your services run stably. After creating an ECS, you can use it like using your local computer or physical server.

This document describes ECS application programming interfaces (APIs), including description, syntax, parameters, and examples. For details about all supported operations, see [API Overview](#).

If you plan to access ECSs through an API, ensure that you are familiar with ECS concepts. For details, see "Service Overview" in the *Elastic Cloud Server User Guide*.

API Calling

ECSs support Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see [Calling APIs](#).

Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. For the endpoints of all services, see [Regions and Endpoints](#).

Constraints

- The number of ECSs that you can create is determined by your quota. To view or increase the quota, see "Quota Adjustment" in *Elastic Cloud Server User Guide*.
- For more constraints, see the API description.

Concepts

- Account

An account is created upon successful signing up. 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

A region is a geographic area in which cloud resources are deployed.

Availability zones (AZs) in the same region can communicate with each other over an intranet, while AZs in different regions are isolated from each other.

Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.

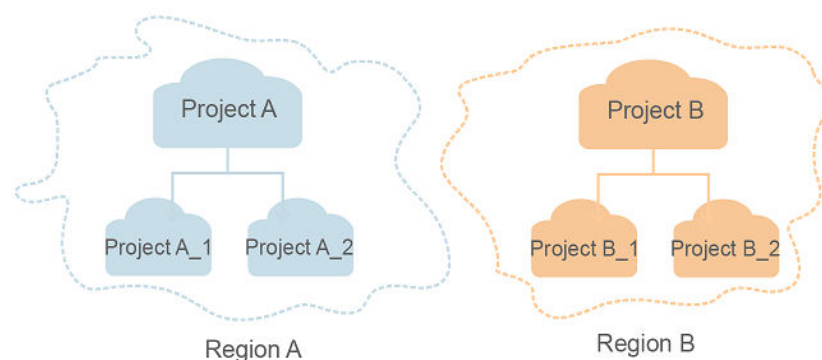
- AZ

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 to group and physically isolate resources (including computing, storage, and network resources) across regions. 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.

Figure 1-1 Project isolation model



2 API Overview

APIs for ECS include native OpenStack APIs and ECS APIs. ECS APIs are recommended.

ECS APIs

Table 2-1 ECS APIs

Type	Description
Lifecycle management	Create, delete, or query ECSs.
Status management	Modify ECS specifications and reinstall or change the ECS OS.
Batch operations	Start, restart, stop, or modify ECSs in a batch.
Flavor management	Query details about flavors and extended flavor information.
NIC management	<ul style="list-style-type: none">• Add or delete ECS NICs in a batch.• Bind or unbind a private IP address to or from an ECS NIC.
Disk management	Attach, detach, or query ECS disks.
Tenant quota management	Query the quotas of a tenant, including the quota limit and used quotas.
Task status management	Query asynchronous API execution status, such as creating or deleting ECSs, performing batch operations on ECSs, or performing operations on NICs.
Tag management	Create, delete, or query ECS D2 tags. Tags can be added or deleted in a batch.

Type	Description
ECS group management	Create or delete an ECS group, add an ECS to an ECS group, or delete an ECS from an ECS group.

Native OpenStack APIs

Table 2-2 Native OpenStack APIs

Type	Description
API version query	<ul style="list-style-type: none"> Query all API versions. Query a specified API version.
Lifecycle management	Create, delete, modify, or query ECSs.
Status management	Start, stop, restart, lock, or unlock ECSs; modify ECS specifications; roll back ECS specifications modification.
Network management	Query ECS tenants or networks.
Image management	Delete or query images. This image management API has been discarded. Use the IMS API.
Security group management	Add, remove, query, create, update, or delete security groups and security group rules.
Flavor management	Query ECS flavors and details.
NIC management	Add, delete, or query ECS NICs.
Disk management	Attach, detach, or query ECS disks.
Metadata management	Update, set, delete, query, obtain, or modify ECS metadata.
Tenant quota management	Query tenant quotas.
Key and password management	Query, create, or delete SSH keys.
Floating IP address management	Allocate, release, create, query, or delete floating IP addresses. This floating IP address management API has been discarded. Use the network service API.

Type	Description
ECS group management	Create, query, or delete ECS groups.
ECS operation management	Query ECS operations or a specified operation by request ID.
ECS console management	Obtain ECS management console logs.
Snapshot management	Create, query, or delete snapshots. The snapshot management API has been discarded. Use the storage service API.
Tag management	Create, delete, or query ECS D1 tags.

3 Selecting an API Type or Version

API Types

ECS APIs are classified as follows:

1. APIs for ECS with customized specifications
2. Native OpenStack APIs that comply with OpenStack community specifications

The two types of APIs offer similar functions but are used in different application scenarios. OpenStack APIs are used to interconnect with open-source ecosystem tools. ECS APIs have enhanced certain functions based on the OpenStack APIs. To better use OpenStack APIs, you are advised to learn about OpenStack concepts and knowledge.

Versions

APIs for ECS include native OpenStack APIs and ECS APIs. You are advised to use ECS APIs.

OpenStack APIs can be of V2 or V2.1. V2.1 supports all functions supported by V2. Additionally, V2.1 supports microversions. If OpenStack APIs are to be used, V2.1 APIs are recommended.

NOTE

To switch an OpenStack API from V2.1 to V2, change **2.1** in the native API URI to **2**.

Microversions

Microversions specify small API changes. A V2.1 API allows you to specify a microversion for related new API functions. To obtain the supported major versions, and maximum and minimum microversions, see [Querying All API Versions](#).

To enable microversion features, add header **X-OpenStack-Nova-API-Version** or **OpenStack-API-Version** to the request when calling an OpenStack API. For example, to enable microversion V2.26 features, add the following header to the HTTPS request:

X-OpenStack-Nova-API-Version: 2.26 or **OpenStack-API-Version: compute 2.26**

 NOTE

If you do not specify the header of a V2.1 API, the system uses header **OpenStack-API-Version: compute 2.1** or **X-OpenStack-Nova-API-Version: 2.1** by default.

Microversion Request Example

For example, you are required to use the API for details about an ECS to view the **OS-EXT-SRV-ATTR:hostname** field.

- **Using a V2 API without a microversion**

- GET: `https://{Endpoint}/v2/74610f3a5ad941998e91f076297ecf27/servers/detail`

{Endpoint} indicates the IAM endpoint. For details, see [Endpoints](#).

- Headers

Content-Type	application/json
X-Auth-Token	\${token}

- Response body

```
{
  "servers": [
    {
      "tenant_id": "74610f3a5ad941998e91f076297ecf27",
      "addresses": {
        "05d4fb93-84e5-4964-853b-32992ffef627": [
          {
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:20:17:95",
            "OS-EXT-IPS:type": "fixed",
            "addr": "192.168.0.228",
            "version": 4
          },
          {
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:20:17:95",
            "OS-EXT-IPS:type": "floating",
            "addr": "192.168.51.61",
            "version": 4
          }
        ]
      },
      "metadata": {},
      "OS-EXT-STS:task_state": null,
      "OS-DCF:diskConfig": "MANUAL",
      "OS-EXT-AZ:availability_zone": "az1-dc1",
      "links": [
        {
          "rel": "self",
          "href": "https://None/v2.1/74610f3a5ad941998e91f076297ecf27/servers/89c312bb-285a-4026-a237-d441908c2f9e"
        },
        {
          "rel": "bookmark",
          "href": "https://None/74610f3a5ad941998e91f076297ecf27/servers/89c312bb-285a-4026-a237-d441908c2f9e"
        }
      ],
      "OS-EXT-STS:power_state": 1,
      "id": "89c312bb-285a-4026-a237-d441908c2f9e",
      "os-extended-volumes:volumes_attached": [
        {
          "id": "c70c4b8e-33bd-4d1f-ab16-14a5a38cdeaf"
        }
      ]
    }
  ]
}
```

```

    }
  ],
  "OS-EXT-SRV-ATTR:host": "pod05.test.01",
  "image": {
    "links": [
      {
        "rel": "bookmark",
        "href": "https://None/74610f3a5ad941998e91f076297ecf27/images/1189efbf-
d48b-46ad-a823-94b942e2a000"
      }
    ],
    "id": "1189efbf-d48b-46ad-a823-94b942e2a000"
  },
  "OS-SRV-USG:terminated_at": null,
  "accessIPv4": "",
  "accessIPv6": "",
  "created": "2018-05-11T03:21:56Z",
  "hostId": "fc7a8ff86bac050f0d9454b1b078dcc97060e819acb06f04c3e338f",
  "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova012@7",
  "key_name": "id_rsa",
  "flavor": {
    "links": [
      {
        "rel": "bookmark",
        "href": "https://None/74610f3a5ad941998e91f076297ecf27/flavors/s3.small.1"
      }
    ],
    "id": "s3.small.1"
  },
  "security_groups": [
    {
      "name": "default"
    }
  ],
  "config_drive": "",
  "OS-EXT-STS:vm_state": "active",
  "OS-EXT-SRV-ATTR:instance_name": "instance-0016c624",
  "user_id": "f79791beca3c48159ac2553fff22e166",
  "name": "zt-test",
  "progress": 0,
  "OS-SRV-USG:launched_at": "2018-05-11T03:22:16.701600",
  "updated": "2018-05-11T03:22:51Z",
  "status": "ACTIVE"
}
]
}

```

- Conclusion: The response body does not contain the **OS-EXT-SRV-ATTR:hostname** field.
- **Using a V2.1 API with a microversion**
 - GET: `https://{Endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers/detail`
{Endpoint} indicates the IAM endpoint. For details, see [Endpoints](#).
 - Headers

Content-Type	application/json
X-Auth-Token	#{token}
X-OpenStack-Nova-API-Version	2.26

- Response body

```
{
  "servers": [
```



```
{
  "tenant_id": "74610f3a5ad941998e91f076297ecf27",
  "addresses": {
    "05d4fb93-84e5-4964-853b-32992ffef627": [
      {
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:20:17:95",
        "OS-EXT-IPS:type": "fixed",
        "addr": "192.168.0.228",
        "version": 4
      },
      {
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:20:17:95",
        "OS-EXT-IPS:type": "floating",
        "addr": "192.168.51.61",
        "version": 4
      }
    ]
  },
  "metadata": {},
  "OS-EXT-STS:task_state": null,
  "description": "zt-test",
  "OS-EXT-SRV-ATTR:hostname": "zt-test",
  "OS-DCF:diskConfig": "MANUAL",
  "OS-EXT-AZ:availability_zone": "az-test-01",
  "links": [
    {
      "rel": "self",
      "href": "https://None/v2.1/74610f3a5ad941998e91f076297ecf27/servers/89c312bb-285a-4026-a237-d441908c2f9e"
    },
    {
      "rel": "bookmark",
      "href": "https://None/74610f3a5ad941998e91f076297ecf27/servers/89c312bb-285a-4026-a237-d441908c2f9e"
    }
  ],
  "OS-EXT-STS:power_state": 1,
  "id": "89c312bb-285a-4026-a237-d441908c2f9e",
  "os-extended-volumes:volumes_attached": [
    {
      "delete_on_termination": true,
      "id": "c70c4b8e-33bd-4d1f-ab16-14a5a38cdeaf"
    }
  ],
  "locked": false,
  "OS-EXT-SRV-ATTR:kernel_id": "",
  "OS-EXT-SRV-ATTR:host": "pod05.test.01",
  "OS-EXT-SRV-ATTR:ramdisk_id": "",
  "image": {
    "links": [
      {
        "rel": "bookmark",
        "href": "https://None/74610f3a5ad941998e91f076297ecf27/images/1189efbf-d48b-46ad-a823-94b942e2a000"
      }
    ]
  },
  "id": "1189efbf-d48b-46ad-a823-94b942e2a000"
},
"accessIPv4": "",
"OS-SRV-USG:terminated_at": null,
"accessIPv6": "",
"OS-EXT-SRV-ATTR:launch_index": 0,
"created": "2018-05-11T03:21:56Z",
"OS-EXT-SRV-ATTR:user_data": null,
"hostId": "fc7a8ff86bac050f0d9454b1b078dcc97060e819acbf06f04c3e338f",
"OS-EXT-SRV-ATTR:reservation_id": "r-pbqmaxer",
"OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
"host_status": "UP",
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova012@7",
```

```
"tags": [],
"key_name": "id_rsa",
"flavor": {
  "links": [
    {
      "rel": "bookmark",
      "href": "https://None/74610f3a5ad941998e91f076297ecf27/flavors/s3.small.1"
    }
  ],
  "id": "s3.small.1"
},
"security_groups": [
  {
    "name": "default"
  }
],
"config_drive": "",
"OS-EXT-STS:vm_state": "active",
"OS-EXT-SRV-ATTR:instance_name": "instance-0016c624",
"user_id": "f79791beca3c48159ac2553fff22e166",
"name": "zt-test",
"progress": 0,
"OS-SRV-USG:launched_at": "2018-05-11T03:22:16.701600",
"updated": "2018-05-11T03:22:51Z",
"status": "ACTIVE"
}
]
```

- Conclusion: The response body contains the **OS-EXT-SRV-ATTR:hostname** field.

Microversion Response Example

If the values of **version** and **min_version** are null, the endpoint does not support microversions.

- **version**: indicates the maximum microversion.
- **min_version**: indicates the minimum microversion.

A microversion on the client must be within the range specified by **version** and **min_version** to access the endpoint. The client uses the following HTTP header to specify a microversion:

```
X-OpenStack-Nova-API-Version: 2.4
```

Since microversion 2.27, the client can also use the following header to specify a microversion:

```
Openstack-API-Version: compute 2.27
```

In the following response example, the maximum microversion is 2.14 and the minimum one is 2.1:

```
{
  "versions": [
    {
      "id": "v2.0",
      "links": [
        {
          "href": "http://openstack.example.com/v2/",
          "rel": "self"
        }
      ]
    },
    {
      "status": "SUPPORTED",
      "version": "",

```

```
    "min_version": "",
    "updated": "2011-01-21T11:33:21Z"
  },
  {
    "id": "v2.1",
    "links": [
      {
        "href": "http://openstack.example.com/v2.1/",
        "rel": "self"
      }
    ],
    "status": "CURRENT",
    "version": "2.14",
    "min_version": "2.1",
    "updated": "2013-07-23T11:33:21Z"
  }
]
}
```

4 Calling APIs

4.1 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 4-1 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 .
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 .
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of <i>Parameter name=Parameter value</i> . For example, ?limit=10 indicates that a maximum of 10 data records will be displayed.

 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 4-2 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://{{endpoint}}/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 4-3 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 <i>Hostname:Port number</i> . 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: MIIPAgYJKoZlhvc NAQcCo...ggg1B BIINPXsidG9rZ

 **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://{{endpoint}}/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 (for example, JSON or XML) 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*, *\$ADMIN_PASS* (login password), and *xxxxxxxxxxxxxxxxxxxx* (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://{{endpoint}}/v3/auth/tokens
Content-Type: application/json
```

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "$ADMIN_PASS", //You are advised to store it in ciphertext in the
configuration file or an environment variable and decrypt it when needed to ensure security.
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

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.

4.2 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.

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

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
```



```
    "user": {
      "name": "username", //IAM user name
      "password": "$ADMIN_PASS", //IAM user password. You are advised to store it in ciphertext
      "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://{{endpoint}}/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.

4.3 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 [HTTP 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

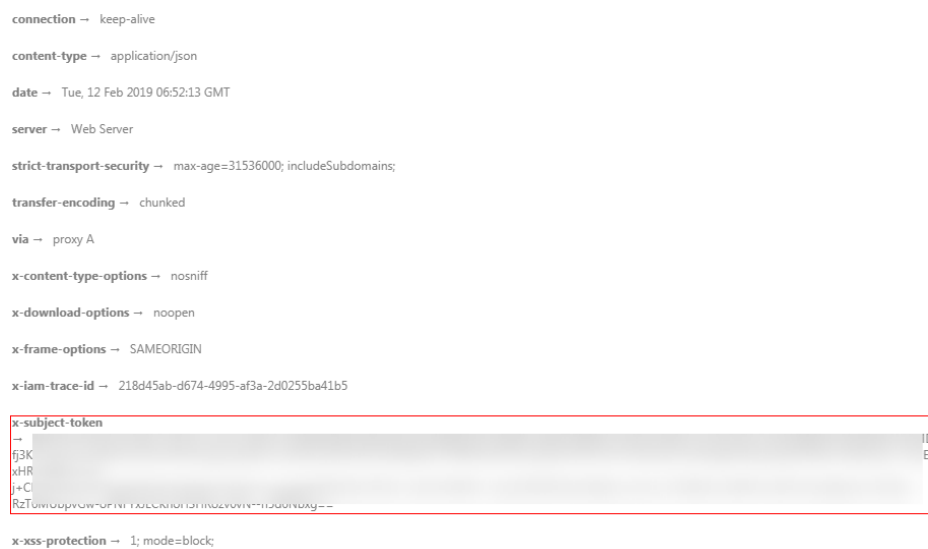
Similar to a request, a response also has a header, for example, **Content-Type**.

Figure 4-1 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.

NOTE

For security purposes, you are advised to set the token in ciphertext in configuration files or environment variables and decrypt it when using it.

Figure 4-1 Header fields of the response to the request for obtaining a user token



```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token → [REDACTED] IC EI
x-xss-protection → 1; mode=block
```

(Optional) Response Body

The body of a response is often returned in a structured format (for example, JSON or XML) 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 request message format is invalid.",
  "error_code": "IMG.0001"
}
```

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

5 APIs (Recommended)

5.1 Lifecycle Management

5.1.1 Creating an ECS

Function

This API is used to create one or more ECSs.

This API is an asynchronous API. After the creation request is successfully delivered, a job ID is returned. This does not mean that the creation is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The **SUCCESS** status indicates that the creation is successful.

Learn how to [authorize and authenticate](#) this API before using it.

Before calling this API, you need to obtain [Regions and Endpoints](#).

Logging in to an ECS can be authenticated using either a key pair or password. For security purposes, you are advised to use key pair authentication.

- Key pair

A key pair is used for ECS login authentication.

Method of calling APIs: Use the **key_name** field to specify the key file used for logging in to the ECS.

- Password

If you choose the initial password for authentication in an ECS, you can log in to the ECS using the username and its initial password. The initial password of user **root** is used for authentication in Linux, while that of user **Administrator** is used for authentication in Windows.

Method of calling APIs: Use the **adminPass** field to specify the initial login password of the administrator account. For details about how to use the **adminPass** field, see [Table 5-3](#). If an encrypted password is required for logging in to a Linux ECS that is created using an image with Cloud-Init installed, you can use the **user_data** field to inject the password. For details, see [Table 5-3](#).

 NOTE

If the **user_data** field is specified for a Linux ECS that is created using an image with Cloud-Init installed, the **adminPass** field becomes invalid.

- Image password

If you use a Linux private image to create an ECS, you can use the image password for login authentication.

Method of calling APIs: If the image password is used, the **key_name** and **adminPass** fields do not need to be specified.

URI

POST /v1/{project_id}/cloudservers

[Table 5-1](#) describes the parameters in the URI.

Table 5-1 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

Request parameters

[Table 5-2](#) describes the request parameters.

Table 5-2 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS information. For details, see Table 5-3 .
dry_run	No	Boolean	Specifies whether to check the request and create the ECS. The default value is false . <ul style="list-style-type: none">• true: The request is sent, but the ECS will not be created. Check items include mandatory parameters and request format.<ul style="list-style-type: none">– If the check fails, the system returns an error.– If the check is successful, the system returns status code 202.• false: The request is sent and the ECS will be created after the check is passed.

Table 5-3 Parameters for creating an ECS

Parameter	Mandatory	Type	Description
imageRef	Yes	String	Specifies the ID of the system image used for creating ECSs. The ID is in Universally Unique Identifier (UUID) format.
flavorRef	Yes	String	Specifies the flavor ID of the ECS to be created. For details about the flavors that have been released, see "ECS Specifications and Types" in the <i>Elastic Cloud Server User Guide</i> .
name	Yes	String	Specifies the ECS name. Value requirements: <ul style="list-style-type: none">• The parameter value consists of 1 to 128 characters, including letters, digits, underscores (_), hyphens (-), and periods (.).• If more than one ECS is to be created (the count value is greater than 1), the system automatically adds a hyphen followed by a four-digit incremental number, such as -0000, to the end of each ECS name. If you specify a number, the name of the first new ECS will start from the specified number. In this case, the ECS name contains a maximum of 59 characters. NOTE ECS hostnames comply with RFC952 and RFC1123 naming rules. It is recommended that you configure hostnames using digits, lower-case letters, and hyphens (-).

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected to the ECS during the creation. Text and text files can be injected.</p> <p>NOTE</p> <ul style="list-style-type: none"> The content of user_data must be encoded with base64. The maximum size of the content to be injected (before encoding) is 32 KB. If the user_data field is specified for a Linux ECS that is created using an image with Cloud-Init installed, the adminPass field becomes invalid. <p>For more information about the user data to be injected, see "Injecting User Data into ECSs" in the <i>Elastic Cloud Server User Guide</i>.</p> <p>Examples</p> <p>Before base64 encoding:</p> <ul style="list-style-type: none"> Linux #!/bin/bash echo user_test > /home/user.txt Windows rem cmd echo 111 > c:\aaa.txt <p>After base64 encoding:</p> <ul style="list-style-type: none"> Linux lyEvYmluL2Jhc2gKZWNObyB1c2VyX3Rlc3QgPiAvaG9tZS91c2VyLnR4dA== Windows cmVtIGNtZApLY2hvIDExMSA+IGM6XGFhYS50eHQ=

Parameter	Mandatory	Type	Description
adminPass	No	String	<p>Specifies the initial login password of the administrator account for logging in to an ECS using password authentication. The Linux administrator is root, and the Windows administrator is Administrator. For details, see Function.</p> <p>Password complexity requirements:</p> <ul style="list-style-type: none">• Consists of 8 to 26 characters.• The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@\$%^&_ = +[{}];,./?~#*).• The password cannot contain the username or the username in reverse.• The Windows ECS password cannot contain the username, the username in reverse, or more than two consecutive characters in the username.
key_name	No	String	<p>Specifies the name of the SSH key used for logging in to the ECS.</p> <p>Keys can be created using the key creation API (Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (Querying SSH Key Pairs).</p>
vpcid	Yes	String	<p>Specifies the ID of the VPC to which the ECS belongs. The value is in the format of the UUID.</p> <p>You can obtain the VPC ID from the management console or by following the instructions provided in "Querying VPCs" in <i>Virtual Private Cloud API Reference</i>.</p>
nics	Yes	Array of objects	<p>Specifies the NIC information of the ECS. For details, see Table 5-4.</p> <p>Constraints:</p> <ul style="list-style-type: none">• The value must be the ID of the subnet created in the VPC specified by vpcid and in the format of the UUID.• A maximum of 12 NICs can be attached to an ECS.

Parameter	Mandatory	Type	Description
publicip	No	Object	Specifies the EIP bound to the ECS, which can be configured in one of the following ways: <ul style="list-style-type: none">Do not use: In such a case, this parameter is unavailable.Automatically assign: You need to specify the information about the EIP to be created.Use existing: You need to specify an existing EIP for your ECS. For details, see Table 8-1 .
count	No	Integer	Specifies the number of ECSs to be created. Constraints: <ul style="list-style-type: none">If this parameter is not specified, the default value is 1.If the quota is sufficient, the maximum value is 500.
root_volume	Yes	Object	Specifies ECS system disk configurations. For details, see Table 5-5 .
data_volumes	No	Array of objects	Specifies ECS data disk configurations. Each data structure represents a data disk to be created. An ECS can be attached with a maximum of 59 data disks (certain flavors support only 23 data disks). For details, see Table 5-6 .
security_groups	No	Array of objects	Specifies the security groups of the ECS. If this parameter is left blank, the default security group is bound to the ECS by default. For details, see Table 8-2 .
availability_zone	No	String	Specifies the name of the AZ where the ECS is located. NOTE If this parameter is not specified, the system automatically selects an AZ. If the ECS is created on a DeH, the ECS must be in the same AZ as the DeH.
extendparam	No	Object	Provides the supplementary information about the ECS to be created. For details, see Table 8-6 .

Parameter	Mandatory	Type	Description
metadata	No	Object	<p>Specifies the metadata of the ECS to be created.</p> <p>You can use metadata to customize key-value pairs.</p> <p>NOTE</p> <ul style="list-style-type: none"> If the metadata contains sensitive data, take appropriate measures to protect the sensitive data, for example, controlling access permissions and encrypting the data. A maximum of 10 key-value pairs can be injected. A metadata key consists of 1 to 255 characters and contains only uppercase letters, lowercase letters, spaces, digits, hyphens (-), underscores (_), colons (:), and decimal points (.). A metadata value consists of a maximum of 255 characters. <p>For details about reserved key-value pairs, see Table 8-8.</p>
os:scheduler_hints	No	Object	<p>Schedules ECSs, for example, by configuring an ECS group.</p> <p>For details, see Table 8-9.</p>
description	No	String	<p>Specifies the description of the ECS, which is empty by default.</p> <ul style="list-style-type: none"> Can contain a maximum of 85 characters. Cannot contain an angle bracket < or >.

Table 5-4 nics field description

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	<p>Specifies the subnet of the ECS.</p> <p>The value must be the ID of the subnet created in the VPC specified by vpcid and in the format of the UUID.</p>

Parameter	Mandatory	Type	Description
ip_address	No	String	Specifies the IP address of the NIC used by the ECS. The value is an IPv4 address. Constraints: <ul style="list-style-type: none">If this parameter is left blank or set to "", an unused IP address in the subnet is automatically assigned as the IP address of the NIC.If this parameter is specified, its value must be an unused IP address in the network segment of the subnet.
binding:profile	No	Object	Allows you to customize data. Configure this parameter when creating an ECS for HANA. For details, see Table 8-11 .
extra_dhcp_opts	No	Array of objects	Indicates extended DHCP options. For details, see Table 8-12 .

Table 5-5 root_volume field description

Parameter	Mandatory	Type	Description
volumetype	Yes	String	Specifies the ECS system disk type, which must be one of available disk types. The value can be SATA or SSD . <ul style="list-style-type: none">SSD: the ultra-high I/O typeSATA: the common I/O type If the specified disk type is not available in the AZ, the disk will fail to be created. NOTE <ul style="list-style-type: none">For details about disk types, see Disk Types and Disk Performance in the <i>Elastic Volume Service User Guide</i>.

Parameter	Mandatory	Type	Description
size	No	Integer	<p>Specifies the system disk size in GB. The value ranges from 1 to 1024.</p> <p>Constraints:</p> <ul style="list-style-type: none"> The system disk size must be greater than or equal to the minimum system disk size supported by the image (min_disk attribute of the image). If this parameter is not specified or is set to 0, the default system disk size is the minimum value of the system disk in the image (min_disk attribute of the image). <p>NOTE To obtain the minimum system disk size (min_disk) of an image, click the image on the management console for its details. Alternatively, call the native OpenStack API for querying details about an image. For details, see "Querying Image Details (Native OpenStack)" in <i>Image Management Service API Reference</i>.</p>
cluster_type	No	String	<p>Specifies the storage type of an ECS system disk.</p> <p>Value: DSS (dedicated storage type)</p> <p>This parameter must be used together with cluster_id and cluster_id must be specified.</p>
cluster_id	No	String	<p>Specifies the ID of the storage pool used by an ECS system disk.</p>
hw:passthrough	No	Boolean	<p>Specifies the device type of the EVS disks to be created.</p> <ul style="list-style-type: none"> If this parameter is set to false, VBD disks are created. If this parameter is set to true, SCSI disks are created. If this parameter is not specified or set to a non-Boolean character, VBD disks are created by default.

Table 5-6 data_volumes field description

Parameter	Mandatory	Type	Description
volumetype	Yes	String	Specifies the type of the ECS data disk, which must be one of available disk types. The value can be SATA or SSD . <ul style="list-style-type: none">• SSD: the ultra-high I/O type• SATA: the common I/O type If the specified disk type is not available in the AZ, the disk will fail to be created. NOTE <ul style="list-style-type: none">• For details about disk types, see Disk Types and Disk Performance in the <i>Elastic Volume Service User Guide</i>.
size	Yes	Integer	Specifies the data disk size, in GB. The value ranges from 10 to 32768. When you use a data disk image to create a data disk, ensure that the value of this parameter is greater than or equal to the size of the source data disk that is used to create the data disk image.
hw:passthrough	No	Boolean	Specifies the device type of the EVS disks to be created. <ul style="list-style-type: none">• If this parameter is set to false, VBD disks are created.• If this parameter is set to true, SCSI disks are created.• If this parameter is not specified or set to a non-Boolean character, VBD disks are created by default.
cluster_type	No	String	Specifies the storage type of an ECS data disk. Value: DSS (dedicated storage type) This parameter must be used together with cluster_id and cluster_id must be specified.
cluster_id	No	String	Specifies the ID of the storage pool used by an ECS data disk.
data_image_id	No	String	Specifies ID of the data image. The value is in UUID format. If data disks are created using a data disk image, this parameter is mandatory and it does not support metadata.

Parameter	Mandatory	Type	Description
metadata	No	Object	<p>Specifies the EVS disk metadata. Ensure that key and value in the metadata contain at most 255 bytes.</p> <p>This field is used only when an encrypted disk is created.</p> <p>If data disks are created using a data disk image, this field cannot be used.</p> <p>For details, see metadata Field Description for Creating Disks.</p>

Response

Parameter	Type	Description
job_id	String	<p>Specifies the returned task ID after delivering the task. You can query the task progress using this ID. For details about how to query the task execution status based on job_id, see Task Status Management.</p>

For details about abnormal responses, see [Responses \(Task\)](#).

Example Request

- Create a pay-per-use ECS with 2 vCPUs, 16 GiB of memory, SSD disks attached, and 10 Mbit/s of bandwidth-billed EIP bound. Use the key pair for login authentication.

```
POST https://{endpoint}/v1/{project_id}/cloudservers
{
  "server": {
    "availability_zone": "az1-dc1",
    "name": "newservers",
    "imageRef": "imageid_123",
    "root_volume": {
      "volumetype": "SSD"
    }
  },
  "data_volumes": [
    {
      "volumetype": "SSD",
      "size": 100
    },
    {
      "volumetype": "SSD",
      "size": 100,
      "multiattach": true,
      "hw:passthrough": true
    }
  ],
  "flavorRef": "m3.large.8",
  "vpcid": "0dae26c9-9a70-4392-93f3-87d53115d171",
  "security_groups": [

```

```
{
  "id": "507ca48f-814c-4293-8706-300564d54620"
},
"nics": [
  {
    "subnet_id": "157ee789-03ea-45b1-a698-76c92660dd83",
    "extra_dhcp_opts": [
      {
        "opt_value": 8888,
        "opt_name": "26"
      }
    ]
  }
],
"publicip": {
  "eip": {
    "iptype": "5_bgp",
    "bandwidth": {
      "size": 10,
      "sharetype": "PER"
    }
  }
},
"key_name": "sshkey-123",
"count": 1
}
```

- Send a pre-verification request to check whether mandatory parameters are configured in the request and whether the request format is correct.

POST https://endpoint/v1/{project_id}/cloudservers

```
{
  "dry_run": true,
  "server": {
    "availability_zone": "az1-dc1",
    "name": "newservers",
    "imageRef": "1189efbf-d48b-46ad-a823-94b942e2a000",
    "root_volume": {
      "volumetype": "SSD"
    }
  },
  "data_volumes": [
    {
      "volumetype": "SSD",
      "size": 100,
      "multiattach": true,
      "hw:passthrough": true
    }
  ],
  "flavorRef": "s3.xlarge.2",
  "vpcid": "0dae26c9-9a70-4392-93f3-87d53115d171",
  "security_groups": [
    {
      "id": "507ca48f-814c-4293-8706-300564d54620"
    }
  ],
  "nics": [
    {
      "subnet_id": "157ee789-03ea-45b1-a698-76c92660dd83"
    }
  ],
  "key_name": "sshkey-123",
  "count": 1
}
```

Example Response

```
{
  "job_id": "93c82933d6b7827d3016b8771f2070873"
}
```

Or

```
{
  "error": {
    "code": "request body is illegal.",
    "message": "Ecs.0005"
  }
}
```

Or

```
{
  "error": {
    "message": "privateIp [%s] is not in this subnet [%s]",
    "code": "Ecs.0005",
    "details": [
      {
        "code": "Ecs.0039"
      }
    ]
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.1.2 Deleting ECSs

Function

This API is used to delete ECSs based on a specified ECS ID list.

This API is an asynchronous API. After the deletion request is successfully delivered, a job ID is returned. This does not mean the deletion is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the deletion is successful.

You can delete a single ECS or multiple ECSs in a batch. A maximum of 1,000 ECSs can be deleted in a batch.

URI

POST /v1/{project_id}/cloudservers/delete

[Table 5-7](#) describes the parameters in the URI.

Table 5-7 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 5-8](#) describes the request parameters.

Table 5-8 Request parameters

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies the ECSs to be deleted. For details, see Table 5-9 .
delete_publicip	No	Boolean	<p>Specifies whether to delete the EIP bound to the ECS when deleting the ECS. If you do not want to delete the EIP, the system only unbinds the EIP from the ECS and reserves the EIP.</p> <p>The value can be true or false.</p> <ul style="list-style-type: none">• true: When an ECS is deleted, the EIP bound to the ECS is also released regardless of whether delete_on_termination of the EIP is true or false.• false: When an ECS is deleted, the EIP is only unbound from the ECS and will not be released regardless of whether delete_on_termination of the EIP is true or false. <p>NOTE If delete_publicip is not specified, the delete_on_termination value of the EIP decides whether the EIP is released when the ECS is deleted.</p> <ul style="list-style-type: none">• If delete_on_termination is true and delete_public is null, the EIP is released when the ECS is deleted.• If delete_on_termination is false and delete_public is null, the EIP is only unbound from the ECS and will not be released when the ECS is deleted.

Parameter	Mandatory	Type	Description
delete_volume	No	Boolean	Specifies whether to delete the data disks attached to an ECS when deleting the ECS. If you set the parameter value to false , the system only detaches the disks from the ECS and reserves the disks. The default value is false . <ul style="list-style-type: none">• true: indicates to delete the data disks attached to the ECS when deleting the ECS.• false: indicates only to detach the data disks attached to the ECS when deleting the ECS.

Table 5-9 servers field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the ID of the ECS to be deleted.

Response

See [Responses \(Task\)](#).

Example Request

Delete the ECS whose ID is **616fb98f-46ca-475e-917e-2563e5a8cd19**, unbind the EIP, and detach data disks.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/delete
```

```
{
  "servers": [
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19"
    }
  ],
  "delete_publicip": false,
  "delete_volume": false
}
```

Example Response

```
{
  "job_id": "ff80808288d415d80189901d8eb81cbb"
}
```

Or

```
{
  "error": {
    "message": "request body is illegal.",
    "code": "Ecs.0005"
  }
}
```

```
}  
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.2 Status Management

5.2.1 Reinstalling an ECS OS (Using an Image with Cloud-Init Installed)

Function

This API is used to reinstall an ECS OS. During the system disk reinstallation using the original image, the data disks of the ECS remain unchanged.

This API is an asynchronous API. After the OS reinstallation request is successfully delivered, a job ID is returned. This does not mean the reinstallation is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the reinstallation is successful.

After this API is called, the system uninstalls the system disk, uses the original image to create a system disk, and attaches it to the ECS. In this way, the OS is reinstalled.

Constraints

- You can only use an image with Cloud-Init or Cloudbase-Init installed.
- You can reinstall OS only on an ECS that is stopped or for which OS reinstallation has failed.
- You are not allowed to reinstall the OS of an ECS that does not have the system disk.
- You are not allowed to perform other operations when reinstalling the OS. Otherwise, reinstalling the OS will fail.

URI

POST /v2/{project_id}/cloudservers/{server_id}/reinstallos

[Table 5-10](#) describes the parameters in the URI.

Table 5-10 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-11](#) describes the request parameters.

Table 5-11 Request parameters

Parameter	Mandatory	Type	Description
os-reinstall	Yes	Object	Reinstalls an ECS OS. For details, see Table 5-12 .

Table 5-12 os-reinstall field description

Parameter	Mandatory	Type	Description
keyname	Yes	String	Specifies the key pair name.
userid	Yes	String	Specifies the user ID.
metadata	No	Object	Specifies metadata of the reinstalled ECS.

Table 5-13 metadata field description

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected to the ECS during the creation. Text and text files can be injected.</p> <p>NOTE</p> <ul style="list-style-type: none"> The content of user_data must be encoded with base64. The maximum size of the content to be injected (before encoding) is 32 KB. <p>For more details, see "Injecting User Data into ECSs" in the <i>Elastic Cloud Server User Guide</i>.</p> <p>Examples</p> <p>Before base64 encoding:</p> <ul style="list-style-type: none"> Linux #!/bin/bash echo user_test > /home/user.txt Windows rem cmd echo 111 > c:\aaa.txt <p>After base64 encoding:</p> <ul style="list-style-type: none"> Linux IyEvYmluL2Jhc2gKZWNoYm91c2VyX3Rlc3QgPiAvaG9tZS91c2VyLnR4dA== Windows cmVtIGNtZApY2hVIDExMSA+IGM6XGFhYS50eHQ=

Response

See [Responses \(Task\)](#).

Example Request

- After the ECS OS is reinstalled, use the password for login authentication. For security purposes, store the password in ciphertext in configuration files or environment variables.

```
POST https://{endpoint}/v2/{project_id}/cloudservers/{server_id}/reinstallos
```

```
{
  "os-reinstall": {
    "userid": "7e25b1da389f4697a79df3a0e5bd494e",
    "mode": "withStopServer"
  }
}
```

- Reinstall an OS and use the key pair for login authentication after the reinstallation.

```
POST https://{endpoint}/v2/{project_id}/cloudservers/{server_id}/reinstallos
```

```
{
  "os-reinstall": {
```

```
"keyname": "KeyPair-350b",  
"userid": "7e25b1da389f4697a79df3a0e5bd494e"  
}
```

Example Response

See [Responses \(Task\)](#).

```
{  
  "job_id": "ff80808288d41e1b018990260955686a"  
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.2.2 Changing an ECS OS (Using an Image with Cloud-Init Installed)

Function

This API is used to change an ECS OS. During the system disk reinstallation using a new image, the data disks of the ECS remain unchanged.

This API is an asynchronous API. After the OS change request is successfully delivered, a job ID is returned. This does not mean the OS change is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the OS change is successful.

After this API is called, the system uninstalls the system disk, uses the new image to create a system disk, and attaches it to the ECS. In this way, the OS is changed.

Constraints

- You can only use an image with Cloud-Init or Cloudbase-Init installed.
- Only a stopped ECS or an ECS on which reinstalling or changing the OS failed supports changing OS.
- Only an ECS with a system disk supports changing OS.
- You are not allowed to perform other operations when changing the OS. Otherwise, changing the OS will fail.

URI

POST /v2/{project_id}/cloudservers/{server_id}/changeos

[Table 5-14](#) describes the parameters in the URI.

Table 5-14 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-15](#) describes the request parameters.

Table 5-15 Request parameters

Parameter	Mandatory	Type	Description
os-change	Yes	Object	Changes an ECS OS. For details, see Table 5-16 .

Table 5-16 os-change field description

Parameter	Mandatory	Type	Description
keyname	Yes	String	Specifies the key pair name.
userid	No	String	Specifies the user ID. When the keyname parameter is being specified, the value of this parameter is used preferentially. If this parameter is left blank, the user ID in the token is used by default.
imageid	Yes	String	Specifies the ID of the new image in UUID format. You can obtain the image ID from the console or by following the instructions provided in "Querying Images" in <i>Image Management Service API Reference</i> .
metadata	No	Object	Specifies the metadata of the ECS for which the OS is to be changed.

Table 5-17 metadata field description

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected to the ECS during the creation. Text and text files can be injected.</p> <p>NOTE</p> <ul style="list-style-type: none">The content of user_data must be encoded with base64.The maximum size of the content to be injected (before encoding) is 32 KB. <p>For more details, see "Injecting User Data into ECSs" in the <i>Elastic Cloud Server User Guide</i>.</p> <p>Examples</p> <p>Before base64 encoding:</p> <ul style="list-style-type: none">Linux #!/bin/bash echo user_test > /home/user.txtWindows rem cmd echo 111 > c:\aaa.txt <p>After base64 encoding:</p> <ul style="list-style-type: none">Linux lyEvYmluL2Jhc2gKZWNObyB1c2VyX3Rlc3QgPiAvaG9tZS91c2VyLnR4dA==Windows cmVtIGNtZApY2hvlDExMSA +IGM6XGFhYS50eHQ=

Response

See [Responses \(Task\)](#).

Example Request

- After the ECS OS is switched, use the password for login authentication. For security purposes, store the password in ciphertext in configuration files or environment variables.

```
POST https://{endpoint}/v2/{project_id}/cloudservers/{server_id}/changeos
```

```
{
  "os-change": {
    "userid": "7e25b1da389f4697a79df3a0e5bd494e",
    "imageid": "e215580f-73ad-429d-b6f2-5433947433b0"
  }
}
```

- Change the OS and use the key pair for login authentication after the OS change.

```
POST https://{endpoint}/v2/{project_id}/cloudservers/{server_id}/changeos
```

```
{
  "os-change": {
```



```
"keyname": "KeyPair-350b",
"useruid": "7e25b1da389f4697a79df3a0e5bd494e",
"imageid": "e215580f-73ad-429d-b6f2-5433947433b0"
}
}
```

Example Response

See [Responses \(Task\)](#).

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.2.3 Cold Migrating an ECS

Function

- An ECS deployed on a DeH can be migrated to another DeH.
- An ECS deployed on a DeH can be migrated to a public resource pool.
- An ECS deployed in a public resource pool can be migrated to a DeH.

This API is an asynchronous API. After the cold migration request is successfully delivered, a job ID is returned. This does not mean the cold migration is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the cold migration is successful.

NOTE

If the migration does not cross NUMA nodes, the migration may fail due to insufficient resources on a single NUMA node.

Constraints

- Currently, this API applies only to dedicated hosts.
- Only a stopped ECS can be cold migrated.
- Existing constraints of the native cold migration API are inherited.

URI

POST /v1/{project_id}/cloudservers/{server_id}/migrate

[Table 5-18](#) describes the parameters in the URI.

Table 5-18 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-19](#) describes the request parameters.

Table 5-19 Request parameters

Parameter	Mandatory	Type	Description
migrate	Yes	Object	Specifies the ECS to be migrated. For details, see Table 5-20 . This parameter is null when you migrate an ECS from a dedicated host to a public resource pool.

Table 5-20 migrate field description

Parameter	Mandatory	Type	Description
dedicated_host_id	No	String	Specifies the DeH ID. This parameter takes effect when an ECS is migrated from a public resource pool to a DeH or between DeHs.

Response

See [Responses \(Task\)](#).

Example Request

Migrate the ECS from the public resource pool to the DeH whose ID is **459a2b9d-804a-4745-ab19-a113bb1b4ddc**.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/migrate
{
  "migrate": {
    "dedicated_host_id": "459a2b9d-804a-4745-ab19-a113bb1b4ddc"
  }
}
```

Example Response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.2.4 Modifying the Specifications of an ECS

Function

ECS specifications can be modified, for example, upgrading the vCPUs and memory, to meet service requirements. This API is used to modify ECS specifications.

This API is an asynchronous API. After the specifications modification request is successfully delivered, a job ID is returned. This does not mean the modification is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the modification is successful.

Constraints

- You can modify the ECS specifications only when the ECS is stopped.

URI

POST /v1/{project_id}/cloudservers/{server_id}/resize

[Table 5-21](#) describes the parameters in the URI.

Table 5-21 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-22](#) describes the request parameters.

Table 5-22 Request parameters

Parameter	Mandatory	Type	Description
resize	Yes	Object	Specifies the operation to modify ECS specifications. For details, see Table 5-23 .
dry_run	No	Boolean	Specifies whether to only check the request and not modify the ECS specifications. true: The request is sent and the ECS specifications are not modified. Check items include mandatory parameters and request format. <ul style="list-style-type: none">• If the check fails, the system returns an error.• If the check is successful, the system returns status code 202. false: The request is sent and the ECS specifications will be modified after the check is passed.

Table 5-23 resize field description

Parameter	Mandatory	Type	Description
flavorRef	Yes	String	Specifies the flavor ID of the ECS after the modification.

Response

See [Responses \(Task\)](#).

Example Request

Change the ECS flavor to c3.15xlarge.2.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/resize
{
  "resize": {
    "flavorRef": "c3.15xlarge.2"
  }
}
```

Example Response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.3 Batch Operations

5.3.1 Starting ECSs in a Batch

Function

This API is used to start ECSs in a batch based on specified ECS IDs. A maximum of 1,000 ECSs can be started in one minute.

This API is an asynchronous API. After the batch start request is successfully delivered, a job ID is returned. This does not mean the batch start is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the batch start is successful.

URI

POST /v1/{project_id}/cloudservers/action

[Table 5-24](#) describes the parameters in the URI.

Table 5-24 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

Table 5-25 Request parameters

Parameter	Mandatory	Type	Description
os-start	Yes	Object	Specifies the operation to start the ECS. For details, see Table 5-26 .

Table 5-26 os-start field description

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies ECS IDs. For details, see Table 5-27 .

Table 5-27 servers field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the ECS ID.

Response

See [Responses \(Task\)](#).

Example Request

Batch start ECSs whose IDs are **616fb98f-46ca-475e-917e-2563e5a8cd19** and **726fb98f-46ca-475e-917e-2563e5a8cd20** with the request parameter set to **os-start**.

POST https://{endpoint}/v1/{project_id}/cloudservers/action

```
{
  "os-start": {
    "servers": [
      {
        "id": "616fb98f-46ca-475e-917e-2563e5a8cd19"
      },
      {
        "id": "726fb98f-46ca-475e-917e-2563e5a8cd20"
      }
    ]
  }
}
```

Example Response

```
{
  "job_id": "ff808082889bd9690189061140c235fe"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.3.2 Restarting ECSs in a Batch

Function

This API is used to restart ECSs in a batch based on specified ECS IDs. A maximum of 1,000 ECSs can be restarted in one minute.

This API is an asynchronous API. After the batch restart request is successfully delivered, a job ID is returned. This does not mean the batch restart is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the batch restart is successful.

URI

POST /v1/{project_id}/cloudservers/action

[Table 5-28](#) describes the parameters in the URI.

Table 5-28 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 5-29](#) describes the request parameters.

Table 5-29 Request parameters

Parameter	Mandatory	Type	Description
reboot	Yes	Object	Specifies the operation to restart the ECS. For details, see Table 5-30 .

Table 5-30 reboot field description

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the type of the restart operation. <ul style="list-style-type: none">• SOFT: normal restart• HARD: forcible restart (hard restart)
servers	Yes	Array of objects	Specifies ECS IDs. For details, see Table 5-31 .

Table 5-31 servers field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the ECS ID.

Response

See [Responses \(Task\)](#).

Example Request

Batch restart ECSs whose IDs are **616fb98f-46ca-475e-917e-2563e5a8cd19** and **726fb98f-46ca-475e-917e-2563e5a8cd20** with the request parameter set to **reboot**.

POST `https://{endpoint}/v1/{project_id}/cloudservers/action`

```
{
  "reboot": {
    "type": "SOFT",
    "servers": [
      {
        "id": "616fb98f-46ca-475e-917e-2563e5a8cd19"
      },
      {
        "id": "726fb98f-46ca-475e-917e-2563e5a8cd20"
      }
    ]
  }
}
```

Example Response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.3.3 Stopping ECSs in a Batch

Function

This API is used to stop ECSs in a batch based on specified ECS IDs. A maximum of 1,000 ECSs can be stopped in one minute.

This API is an asynchronous API. After the batch stop request is successfully delivered, a job ID is returned. This does not mean the batch stop is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the batch stop is successful.

URI

POST /v1/{project_id}/cloudservers/action

[Table 5-32](#) describes the parameters in the URI.

Table 5-32 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

Table 5-33 Request parameters

Parameter	Mandatory	Type	Description
os-stop	Yes	Object	Specifies the operation to stop the ECS. For details, see Table 5-34 .

Table 5-34 os-stop field description

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies ECS IDs. For details, see Table 5-35 .
type	No	String	Specifies an ECS stop type. The default value is SOFT . SOFT : normal ECS stop (default) HARD : forcible ECS stop

Table 5-35 servers field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the ECS ID.

Response

See [Responses \(Task\)](#).

Example Request

Batch stop ECSs whose IDs are **616fb98f-46ca-475e-917e-2563e5a8cd19** and **726fb98f-46ca-475e-917e-2563e5a8cd20** with the request parameter set to **os-stop**.

POST https://{endpoint}/v1/{project_id}/cloudservers/action

```
{
  "os-stop": {
    "type": "HARD",
    "servers": [
      {
        "id": "616fb98f-46ca-475e-917e-2563e5a8cd19"
      },
      {
        "id": "726fb98f-46ca-475e-917e-2563e5a8cd20"
      }
    ]
  }
}
```

Example Response

```
{
  "job_id": "ff808082889bd9690189061140c235fe"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.4 Flavor Management

5.4.1 Querying Details About Flavors and Extended Flavor Information

Function

This API is used to query details about ECS flavors and extended flavor information.

URI

GET /v1/{project_id}/cloudservers/flavors?availability_zone={availability_zone}

[Table 5-36](#) describes the parameters in the URI.

Table 5-36 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Table 5-37 Query parameters

Parameter	Mandatory	Type	Description
availability_zone	No	String	Specifies an AZ. If this parameter is not left blank, flavors in the normal , obt , or promotion states are returned.

Request

None

Response

[Table 5-38](#) describes the response parameters.

Table 5-38 Response parameters

Parameter	Type	Description
flavors	Array of objects	Specifies ECS flavors. For details, see Table 5-39 .

Table 5-39 flavors field description

Parameter	Type	Description
id	String	Specifies the ID of the ECS flavor.
name	String	Specifies the name of the ECS flavor.
vcpus	String	Specifies the number of vCPUs in the ECS flavor.
ram	Integer	Specifies the memory size (MB) in the ECS flavor.
disk	String	Specifies the system disk size in the ECS flavor. This parameter has not been used. Its default value is 0 .

Parameter	Type	Description
swap	String	Specifies the swap partition size required by the ECS flavor. This parameter has not been used. Its default value is "".
OS-FLV-EXT-DATA:ephemeral	Integer	Specifies the temporary disk size. This is an extended attribute. This parameter has not been used. Its default value is 0 .
OS-FLV-DISABLED:disabled	Boolean	Specifies whether the ECS flavor has been disabled. This is an extended attribute. This parameter has not been used. Its default value is false .
rxtx_factor	Float	Specifies the ratio of the available network bandwidth to the network hardware bandwidth of the ECS. This parameter has not been used. Its default value is 1.0 .
rxtx_quota	String	Specifies the software constraints of the network bandwidth that can be used by the ECS. This parameter has not been used. Its default value is null .
rxtx_cap	String	Specifies the hardware constraints of the network bandwidth that can be used by the ECS. This parameter has not been used. Its default value is null .
os-flavor-access:is_public	Boolean	Specifies whether a flavor is available to all tenants. This is an extended attribute. <ul style="list-style-type: none">• true: indicates that a flavor is available to all tenants.• false: indicates that a flavor is available only to certain tenants. Default value: true
links	Array of objects	Specifies shortcut links for ECS flavors. For details, see Table 5-40 .
os_extra_specs	Object	Specifies extended ECS specifications. For details, see Table 5-41 .

Table 5-40 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Provides the corresponding shortcut link.
type	String	Specifies the shortcut link type. This parameter has not been used. Its default value is null .

Table 5-41 os_extra_specs field description

Parameter	Type	Description
ecs:performancetype	String	Specifies the ECS flavor type: <ul style="list-style-type: none">• normal: general computing• computingv3: general computing-plus• highmem: memory-optimized• cpuv1: computing I• cpuv2: computing II• highcpu: high-performance computing• diskintensive: disk-intensive• gpu: GPU-accelerated• saphana: large-memory
resource_type	String	Specifies the resource type. resource_type is used to differentiate between the types of the physical servers accommodating ECSs.
instance_vnic:type	String	Specifies the NIC type. The value of this parameter is consistently enhanced , indicating that network enhancement ECSs are to be created.
instance_vnic:instance_bandwidth	String	Specifies the maximum bandwidth in the unit of Mbit/s. The maximum value of this parameter is 10,000 .
instance_vnic:max_count	String	Specifies the maximum number of NICs. The maximum value of this parameter is 4.

Parameter	Type	Description
quota:local_disk	String	<p>The value of this parameter is in format of {type}:{count}:{size}:{safeFormat}.</p> <ul style="list-style-type: none">● type: indicates the disk type, which can only be HDD.● count: indicates the number of local disks. The following types are supported:<ul style="list-style-type: none">- For D1 ECSs, the value can be 3, 6, 12, or 24.- For D2 ECSs, the value can be 2, 4, 8, 12, 16, or 24.- For D3 ECSs, the value can be 2, 4, 8, 12, 16, 24, or 28.● size: indicates the capacity of a single disk, in GB. Currently, only 1675 is supported. The actual disk size is 1800, and the available size after formatting is 1675.● safeFormat: indicates whether the local disks of the ECS are securely formatted. The following types are supported:<ul style="list-style-type: none">- For D1 ECSs, the value is FALSE.- For D2 or D3 ECSs, the value is True. <p>NOTE This field is dedicated for disk-intensive ECSs.</p>
quota:nvme_ssd	String	<p>The value of this parameter is in the format of {type}:{spec}:{num}:{size}:{safeFormat}.</p> <ul style="list-style-type: none">● type: indicates the capacity of a single NVME SSD disk attached to the ECS, which can only be 1.6 TB or 3.2 TB.● spec: indicates the specifications of the NVMe SSD disk, which can be large (large specifications) or lvs (small specifications). If spec is set to large, only I series (for example, I3) is supported.● num indicates the number of local disks.● size: indicates the capacity, in the unit of GB, of the disk used by the guest user. If the spec value is large, the value of this parameter is the size of a single disk attached to the ECS. If the value of spec is lvs, the value of size is an integer multiple of 50.● safeFormat: indicates whether the local disks of the ECS are securely formatted. If safeFormat is set to True, only I series (for example, I3) is supported. <p>NOTE This field is dedicated for ultra-high I/O ECSs.</p>

Parameter	Type	Description
extra_spec:io:persistent_grant	String	<p>Specifies whether persistence is supported. The value of this parameter is true.</p> <p>This parameter indicates that the ECS is persistently authorized to access the storage.</p> <p>NOTE This field is dedicated for disk-intensive D1 ECSs.</p>
ecs:generation	String	<p>Specifies the generation of an ECS type.</p> <p>For example, 3 in s3 indicates the general-purpose third-generation ECSs. For details about flavors and generations, see "ECS Specifications" in the <i>Elastic Cloud Server User Guide</i>.</p>
ecs:virtualization_env_types	String	<p>Specifies a virtualization type.</p> <ul style="list-style-type: none">• If the parameter value is FusionCompute, the ECS uses Xen virtualization.• If the parameter value is CloudCompute, the ECS uses KVM virtualization. <p>NOTE This field is optional.</p>
pci_passthrough:enable_gpu	String	<p>Specifies whether the GPU is passthrough.</p> <p>If the value is true, the GPU is passthrough.</p>
pci_passthrough:gpu_specs	String	<p>Specifies the technology used on the G1 and G2 ECSs, including GPU virtualization and GPU passthrough.</p> <ul style="list-style-type: none">• If the ECS with specified specifications uses GPU virtualization and the GPU model is M60-1Q, the parameter value can be m60_1q:virt:1.• If the ECS with specified specifications uses GPU passthrough and the GPU model is M60, the parameter value can be m60:direct_graphics:1.

Parameter	Type	Description
cond:operation:status	String	<p>This parameter takes effect region-wide. If an AZ is not configured in the cond:operation:az parameter, the value of this parameter is used by default. If this parameter is not set or used, the meaning of normal applies. Options:</p> <ul style="list-style-type: none">• normal: indicates normal commercial use of the flavor.• abandon: indicates that the flavor has been taken offline (not displayed).• sellout: indicates that the flavor has been sold out.• obt: indicates that the flavor is under open beta testing (OBT).• obt_sellout: indicates that the OBT resources are sold out.• promotion: indicates that the flavor is recommended (for commercial use, which is similar to normal).
cond:operation:az	String	<p>This parameter takes effect AZ-wide. If an AZ is not configured in this parameter, the value of the cond:operation:status parameter is used by default. This parameter is in the format of "az(xx)". The value in parentheses is the flavor status in an AZ. If the parentheses are left blank, the configuration is invalid. The cond:operation:az options are the same as the cond:operation:status options.</p> <p>For example, a flavor is for commercial use in AZs 0 and 3, sold out in AZ 1, for OBT in AZ 2, and is canceled in other AZs. Then, set parameters as follows:</p> <ul style="list-style-type: none">• cond:operation:status: abandon• cond:operation:az: az0(normal), az1(sellout), az2(obt), az3(normal) <p>NOTE Configure this parameter if the flavor status in an AZ is different from the cond:operation:status value.</p>
quota:max_rate	String	<p>Specifies the maximum bandwidth.</p> <ul style="list-style-type: none">• Unit: Mbit/s. If a bandwidth is in the unit of Gbit/s, it must be divided by 1,000.
quota:min_rate	String	<p>Specifies the assured bandwidth.</p> <ul style="list-style-type: none">• Unit: Mbit/s. If a bandwidth is in the unit of Gbit/s, it must be divided by 1,000.
quota:max_pps	String	<p>Specifies the maximum intranet PPS.</p> <ul style="list-style-type: none">• Unit: number. If a value is in the unit of 10,000, it must be divided by 10,000.

Parameter	Type	Description
cond:operation:charge	String	Specifies a billing type. <ul style="list-style-type: none">All the billing types are supported if this parameter is not set.
cond:compute	String	Specifies computing constraints. <ul style="list-style-type: none">autorecovery: indicates that automatic recovery is supported.If this parameter does not exist, automatic recovery is not supported.

 NOTE

For more information, see "ECS Specifications and Types" in the *Elastic Cloud Server User Guide*.

Example Request

Query details about ECS flavors and extended flavor information.

```
GET https://{endpoint}/v1/{project_id}/cloudservers/flavors?availability_zone=availability_value
```

Example Response

```
{
  "flavors": [
    {
      "id": "c3.2xlarge.2",
      "name": "c3.2xlarge.2",
      "vcpus": "8",
      "ram": 16384,
      "disk": "0",
      "swap": "",
      "links": [
        {
          "rel": "self",
          "href": "https://ecs.region.example.com/v1.0/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",
          "type": null
        },
        {
          "rel": "bookmark",
          "href": "https://ecs.region.example.com/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",
          "type": null
        }
      ]
    },
    {
      "OS-FLV-EXT-DATA:ephemeral": 0,
      "rxtx_factor": 1,
      "OS-FLV-DISABLED:disabled": false,
      "rxtx_quota": null,
      "rxtx_cap": null,
      "os-flavor-access:is_public": true,
      "os_extra_specs": {
        "ecs:virtualization_env_types": "CloudCompute",
        "ecs:generation": "c3",
        "ecs:performancetype": "computingv3",
        "resource_type": "IOptimizedC3_2"
      }
    }
  ]
}
```

```
}  
  ]  
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.5 NIC Management

5.5.1 Adding NICs to an ECS in a Batch

Function

This API is used to add one or multiple NICs to an ECS.

This API is an asynchronous API. After the NIC addition request is successfully delivered, a job ID is returned. This does not mean the NIC addition is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the NIC addition is successful.

URI

POST /v1/{project_id}/cloudservers/{server_id}/nics

[Table 5-42](#) describes the parameters in the URI.

Table 5-42 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-43](#) describes the request parameters.

Table 5-43 Request parameters

Parameter	Mandatory	Type	Description
nics	Yes	Array of objects	Specifies the NICs to be added. For details, see Table 5-44 .

Table 5-44 nics field description

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	Specifies the information about the NICs to be added to an ECS. The value must be the ID of a created network in UUID format.
security_groups	No	Array of objects	Specifies the security groups for NICs. For details, see Table 5-45 .
ip_address	No	String	Specifies the IP address. If this parameter is unavailable, the IP address is automatically assigned.

Table 5-45 security_groups field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the ID of the security group.

Response

See [Responses \(Task\)](#).

Example Request

Add the NIC whose ID is **d32019d3-bc6e-4319-9c1d-6722fc136a23** and security group ID is **f0ac4394-7e4a-4409-9701-ba8be283dbc3** to an ECS.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/nics
```

```
{
  "nics": [
    {
      "subnet_id": "d32019d3-bc6e-4319-9c1d-6722fc136a23",
      "security_groups": [
        {
          "id": "f0ac4394-7e4a-4409-9701-ba8be283dbc3"
        }
      ]
    }
  ]
}
```

Example Response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.5.2 Deleting NICs from an ECS in a Batch

Function

This API is used to uninstall and delete one or multiple NICs from an ECS.

This API is an asynchronous API. After the deletion request is successfully delivered, a job ID is returned. This does not mean the deletion is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the deletion is successful.

Constraints

The primary NIC of an ECS has routing rules configured and cannot be deleted.

URI

POST /v1/{project_id}/cloudservers/{server_id}/nics/delete

[Table 5-46](#) describes the parameters in the URI.

Table 5-46 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-47](#) describes the request parameters.

Table 5-47 Request parameters

Parameter	Mandatory	Type	Description
nics	Yes	Array of objects	Specifies the NICs to be deleted. For details, see Table 5-48 .

Table 5-48 nics field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the port ID of the NIC. NOTE When the ID is the same as the ECS primary NIC ID, the system will return error code 403.

Response

See [Responses \(Task\)](#).

Example Request

Delete the NIC whose ID is **d32019d3-bc6e-4319-9c1d-6722fc136a23** from an ECS.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/nics/delete
{
  "nics": [
    {
      "id": "d32019d3-bc6e-4319-9c1d-6722fc136a23"
    }
  ]
}
```

Example Response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.6 Disk Management

5.6.1 Attaching a Disk to an ECS

Function

This API is used to attach a disk to an ECS.

This API is an asynchronous API. After the attachment request is successfully delivered, a job ID is returned. This does not mean the attachment is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the attachment is successful.

URI

POST /v1/{project_id}/cloudservers/{server_id}/attachvolume

[Table 5-49](#) describes the parameters in the URI.

Table 5-49 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

Table 5-50 Request parameters

Parameter	Mandatory	Type	Description
volumeAttachment	Yes	Object	Specifies the ECS attachment information. For details, see Table 5-51 .
dry_run	No	Boolean	Specifies whether to check the request and attach the disk. <ul style="list-style-type: none">true: indicates that only the request is sent, and no disk will be attached. Check items include mandatory parameters, request format, and service restrictions. If the check fails, the system returns an error. If the check result is as expected, the system properly responds.false: indicates that only the request is sent and the disk will be attached if the check result is as expected. The default value is false .

Table 5-51 volumeAttachment field description

Parameter	Mandatory	Type	Description
volumeld	Yes	String	Specifies the ID of the disk to be attached. The value is in UUID format.
device	No	String	Indicates the disk device name. NOTE <ul style="list-style-type: none">The new disk device name cannot be the same as an existing one.This parameter is mandatory for Xen ECSs. Set the parameter value to /dev/sda for the system disks of such ECSs and to /dev/sdx for data disks, where x is a letter in alphabetical order. For example, if there are two data disks, set the device names of the two data disks to /dev/sdb and /dev/sdc, respectively. If you set a device name starting with /dev/vd, the system uses /dev/sd by default.For KVM ECSs, set the parameter value to /dev/vda for system disks. The device names for data disks of KVM ECSs are optional. If the device names of data disks are required, set them in alphabetical order. For example, if there are two data disks, set the device names of the two data disks to /dev/vdb and /dev/vdc, respectively. If you set a device name starting with /dev/sd, the system uses /dev/vd by default.For ECSs that only support SCSI disks, set the device name of the system disk to /dev/sda and the device names of data disks in alphabetical order, for example, /dev/sdb and /dev/sdc. The system will not change the default device names.
volume_type	No	String	Specifies the disk type. If volumeld is unavailable and dry_run is set to true , volume_type is available and must be specified.
count	No	Integer	Specifies the number of disks. If volumeld is unavailable and dry_run is set to true , count is available. If count is unavailable, the number of disks is 1 by default.

Parameter	Mandatory	Type	Description
hw:passthrough	No	String	<ul style="list-style-type: none">If this parameter is set to true, the disk device type is SCSI, which allows ECS OSs to directly access the underlying storage media. SCSI reservation commands are supported.If this parameter is set to false, the disk device type is VBD, which supports only simple SCSI read/write commands. If volumeld is unavailable and dry_run is set to true , volume_type is available and must be specified.

Response

See [Responses \(Task\)](#).

Example Request

Attach a SCSI EVS disk to the device `/dev/sda`.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/attachvolume
{
  "volumeAttachment": {
    "volumeld": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
    "device": "/dev/sda",
    "volume_type": "SSD",
    "count": 5,
    "hw:passthrough": "true"
  },
  "dry_run": false
}
```

Example Response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.6.2 Detaching an EVS Disk from an ECS

Function

This API is used to detach an EVS disk from an ECS.

This API is an asynchronous API. After the detachment request is successfully delivered, a job ID is returned. This does not mean the detachment is complete. You need to call the API by referring to [Querying Task Execution Status](#) to query the job status. The SUCCESS status indicates that the detachment is successful.

URI

```
DELETE /v1/{project_id}/cloudservers/{server_id}/detachvolume/{volume_id}?  
delete_flag=0
```

[Table 5-52](#) describes the parameters in the URI.

Table 5-52 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
volume_id	Yes	Specifies the disk ID.
delete_flag	No	Indicates whether to forcibly detach a data disk. <ul style="list-style-type: none">• If yes, set it to 1.• If no, set it to 0. It is set to 0 by default.

Request

None

Response

See [Responses \(Task\)](#).

Example Request

Detach a specified disk from an ECS.

```
DELETE https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/detachvolume/{volume_id}
```

Example Response

```
{  
  "job_id": "ff80808288d41e1b018990260955686a"  
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.6.3 Querying Disk Attachment of an ECS

Function

This API is used to query disk attachment of an ECS.

URI

GET /v2.1/servers/{server_id}/block_device

[Table 5-53](#) describes the parameters in the URI.

Table 5-53 Parameter description

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID in UUID format.

Request

None

Response

[Table 5-54](#) describes the response parameters.

Table 5-54 Response parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies the disks attached to an ECS. For details, see Table 5-55 .

Table 5-55 volumeAttachments parameters

Parameter	Type	Description
serverId	String	Specifies the ECS ID in UUID format.
volumeId	String	Specifies the EVS disk ID in UUID format.
id	String	Specifies the mount ID, which is the same as the EVS disk ID. The value is in UUID format.
size	Integer	Specifies the EVS disk size in GB.

Parameter	Type	Description
device	String	Specifies the drive letter of the EVS disk, which is the device name of the EVS disk.
pciAddress	String	Specifies the PCI address.
bootIndex	Boolean	Specifies the EVS disk boot sequence. <ul style="list-style-type: none">• 0 indicates the system disk.• A non-zero value indicates a data disk.
bus	String	Indicates the disk bus type. Options: virtio and scsi

Example Request

```
GET https://{endpoint}/v2.1/servers/4d8c3732-a248-40ed-bebc-539a6ffd25c0/block_device
```

Example Response

```
{
  "volumeAttachments": [
    {
      "pciAddress": "0000:02:01.0",
      "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
      "device": "/dev/vda",
      "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
      "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
      "size": "40",
      "bootIndex": 0,
      "bus": "virtio"
    },
    {
      "pciAddress": "0000:02:02.0",
      "volumeId": "a26887c6-c47b-4654-abb5-asdf234r234r",
      "device": "/dev/vdb",
      "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
      "id": "a26887c6-c47b-4654-abb5-asdf234r234r",
      "size": "10",
      "bootIndex": 1,
      "bus": "virtio"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.6.4 Querying a Single Disk Attached to an ECS

Function

This API is used to query a disk attached to an ECS.

URI

GET /v2.1/servers/{server_id}/block_device/{volume_id}

[Table 5-56](#) describes the parameters in the URI.

Table 5-56 Parameter description

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID in UUID format.
volume_id	Yes	Specifies the EVS disk ID in UUID format.

Request

None

Response

[Table 5-57](#) describes the response parameters.

Table 5-57 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disk attached to an ECS. For details, see Table 5-58 .

Table 5-58 volumeAttachment parameters

Parameter	Type	Description
serverId	String	Specifies the ECS ID in UUID format.
volumeId	String	Specifies the EVS disk ID in UUID format.
id	String	Specifies the mount ID, which is the same as the EVS disk ID. The value is in UUID format.
size	Integer	Specifies the EVS disk size in GB.
device	String	Specifies the drive letter of the EVS disk, which is the device name of the EVS disk.
pciAddress	String	Specifies the PCI address.

Parameter	Type	Description
bootIndex	Boolean	Specifies the EVS disk boot sequence. <ul style="list-style-type: none">• 0 indicates the system disk.• A non-zero value indicates a data disk.
bus	String	Specifies the disk bus type. Options: virtio and scsi

Example Request

```
GET https://{endpoint}/v2.1/servers/{server_id}/block_device/{volume_id}
```

Example Response

```
{
  "volumeAttachment": {
    "pciAddress": "0000:02:01.0",
    "volumeId": "a26887c6-c47b-4654-abb5-asdf234r234r",
    "device": "/dev/vda",
    "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
    "id": "a26887c6-c47b-4654-abb5-asdf234r234r",
    "size": "40",
    "bootIndex": 0,
    "bus": "virtio"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.7 Tenant Quota Management

5.7.1 Querying Tenant Quotas

Function

This API is used to query the quotas of all resources for a specified tenant, including used quotas.

URI

```
GET /v1/{project_id}/cloudservers/limits
```

[Table 5-59](#) describes the parameters in the URI.

Table 5-59 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

[Table 5-60](#) describes the response parameters.

Table 5-60 Response parameters

Parameter	Type	Description
absolute	Object	Specifies tenant quotas. For details, see Table 5-61 .

Table 5-61 absolute field description

Parameter	Type	Description
maxTotalInstances	Integer	Specifies the maximum number of ECSs that can be requested.
maxTotalCores	Integer	Specifies the maximum number of CPU cores that the current tenant can apply for.
maxTotalRAMSize	Integer	Specifies the maximum memory space (MB) you can use.
maxTotalKeyPairs	Integer	Specifies the maximum number of SSH key pairs you can use.
maxServerMetadata	Integer	Specifies the maximum length of the metadata you can use.
maxPersonality	Integer	Specifies the maximum number of files that can be injected.
maxPersonalitySize	Integer	Specifies the maximum size (byte) of the file to be injected.
maxServerGroups	Integer	Specifies the maximum number of server groups.
maxServerGroupMembers	Integer	Specifies the maximum number of ECSs in an ECS group.

Parameter	Type	Description
totalServerGroupsUsed	Integer	Specifies the number of used server groups.
maxSecurityGroups	Integer	Specifies the maximum number of security groups you can use. NOTE The quota complies with the VPC quota limit.
maxSecurityGroupRules	Integer	Specifies the maximum number of security group rules that you can configure in a security group. NOTE The quota complies with the VPC quota limit.
maxTotalFloatingIps	Integer	Specifies the maximum number of floating IP addresses you can use.
maxImageMeta	Integer	Specifies the maximum length of the image metadata.
totalInstancesUsed	Integer	Specifies the number of used ECSs.
totalCoresUsed	Integer	Specifies the number of the used CPU cores.
totalRAMUsed	Integer	Specifies the used memory size (MB).
totalSecurityGroupsUsed	Integer	Specifies the number of used security groups.
totalFloatingIpsUsed	Integer	Specifies the number of used floating IP addresses.

Example Request

Query the quotas of all resources in a project for a tenant.

```
GET https://{endpoint}/v1/{project_id}/cloudservers/limits
```

Example Response

Example response

```
{
  "absolute":{
    "maxServerMeta":128,
    "maxPersonality":5,
    "maxImageMeta":128,
    "maxPersonalitySize":10240,
    "maxSecurityGroupRules":20,
    "maxTotalKeypairs":-1,
    "totalRAMUsed":75776,
    "totalInstancesUsed":21,
    "maxSecurityGroups":10,
    "totalFloatingIpsUsed":0,
    "maxTotalCores":20480,
```

```
"totalSecurityGroupsUsed":1,  
"maxTotalFloatingIps":10,  
"maxTotalInstances":2048,  
"totalCoresUsed":40,  
"maxTotalRAMSize":25165824,  
"maxServerGroups":10,  
"maxServerGroupMembers":16,  
"totalServerGroupsUsed":2  
}  
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.8 Task Status Management

5.8.1 Querying Task Execution Status

Function

This API is used to query the execution status of an asynchronous request task.

After an asynchronous request task is issued, for example, creating or deleting an ECS, performing operations on ECSs in a batch, or performing operations on ECS NICs, a task ID will be returned, based on which you can query the execution status of the task.

For details about how to obtain the task ID, see [Responses \(Task\)](#).

URI

GET /v1/{project_id}/jobs/{job_id}

[Table 5-62](#) describes the parameters in the URI.

Table 5-62 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
job_id	Yes	Specifies the ID of an asynchronous request task.

Request

None

Response

[Table 5-63](#) describes the response parameters.

Table 5-63 Response parameters

Parameter	Type	Description
status	String	Specifies the task status. <ul style="list-style-type: none">● SUCCESS: indicates the task is successfully executed.● RUNNING: indicates that the task is in progress.● FAIL: indicates that the task failed.● INIT: indicates that the task is being initialized.
entities	Object	Specifies the object of the task. The value of this parameter varies depending on the type of the task. If the task is an ECS-related operation, the value is server_id . If the task is a NIC operation, the value is nic_id . If a sub-Job is available, details about the sub-job are displayed. For details, see Table 5-64 .
job_id	String	Specifies the ID of an asynchronous request task.
job_type	String	Specifies the type of an asynchronous request task.
begin_time	String	Specifies the time when the task started.
end_time	String	Specifies the time when the task finished.
error_code	String	Specifies the returned error code when the task execution fails. After the task is executed successfully, the value of this parameter is null.
fail_reason	String	Specifies the cause of the task execution failure. After the task is executed successfully, the value of this parameter is null.
message	String	Specifies the error message returned when an error occurs in the request to query a task.
code	String	Specifies the error code returned when an error occurs in the request to query a task. For details about the error code, see Returned Values for General Requests .

Table 5-64 entities field description

Parameter	Type	Description
server_id	String	If the task is an ECS-related operation, the value is server_id .
nic_id	String	If the task is a NIC-related operation, the value is nic_id .
sub_jobs_total	Integer	Specifies the number of subtasks.
sub_jobs	Array of objects	Specifies the execution information of a subtask. For details, see Table 5-65 .

Table 5-65 sub_jobs field description

Parameter	Type	Description
status	String	Specifies the task status. <ul style="list-style-type: none">• SUCCESS: indicates the task is successfully executed.• RUNNING: indicates that the task is in progress.• FAIL: indicates that the task failed.• INIT: indicates that the task is being initialized.
entities	Object	Specifies the object of the task. The value of this parameter varies depending on the type of the task. If the task is an ECS-related operation, the value is server_id . If the task is a NIC operation, the value is nic_id . For details, see Table 5-66 .
job_id	String	Specifies the subtask ID.
job_type	String	Specify the subtask type.
begin_time	String	Specifies the time when the task started.
end_time	String	Specifies the time when the task finished.
error_code	String	Specifies the returned error code when the task execution fails. After the task is executed successfully, the value of this parameter is null.
fail_reason	String	Specifies the cause of the task execution failure. After the task is executed successfully, the value of this parameter is null.

Table 5-66 sub_jobs.entities field description

Parameter	Type	Description
server_id	String	If the task is an ECS-related operation, the value is server_id .
nic_id	String	If the task is a NIC-related operation, the value is nic_id .
errorcode_message	String	Indicates the cause of a subtask execution failure.

Example Request

Query the execution status of a specified asynchronous request task.

```
GET https://{endpoint}/v1/{project_id}/jobs/{job_id}
```

Example Response

```
{
  "status": "SUCCESS",
  "entities": {
    "sub_jobs_total": 1,
    "sub_jobs": [
      {
        "status": "SUCCESS",
        "entities": {
          "server_id": "bae51750-0089-41a1-9b18-5c777978ff6d"
        },
        "job_id": "2c9eb2c5544cbf6101544f0635672b60",
        "job_type": "createSingleServer",
        "begin_time": "2016-04-25T20:04:47.591Z",
        "end_time": "2016-04-25T20:08:21.328Z",
        "error_code": null,
        "fail_reason": null
      }
    ]
  },
  "job_id": "2c9eb2c5544cbf6101544f0602af2b4f",
  "job_type": "createServer",
  "begin_time": "2016-04-25T20:04:34.604Z",
  "end_time": "2016-04-25T20:08:41.593Z",
  "error_code": null,
  "fail_reason": null
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.9 Tag Management

5.9.1 Tag Types

Tag management APIs are classified as the APIs for one-dimensional (1D) tags and the APIs for two-dimensional (2D) tags.

- A 1D tag contains a string. All APIs for 1D tags are native OpenStack APIs. For details, see [Tag Management](#).
- A 2D tag consists of a key and a value. All APIs for 2D tags are ECS APIs. For details, see this section.

NOTE

- Use the APIs of the same type to add, delete, modify, or query tags.
- 2D tags are recommended.

5.9.2 Querying ECSs by Tag

Function

This API is used to filter ECSs by tag and obtain all tags of an ECS.

URI

POST /v1/{project_id}/servers/resource_instances/action

[Table 5-67](#) describes the parameters in the URI.

Table 5-67 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 5-68](#) describes the request parameters.

Table 5-68 Request parameters

Parameter	Mandatory	Type	Description
tags	No	Array of objects	Displays the ECSs with all the specified tags. For details, see Table 5-69 . <ul style="list-style-type: none">The structure body must be included.This field cannot be left blank.A key must be unique.Values of the same key must be unique.
not_tags	No	Array of strings	Displays the ECSs with none of specified tags. <ul style="list-style-type: none">The structure body must be included.This field cannot be left blank.Keys must be unique.Values of the same key must be unique.
limit	No	String	Limits the maximum number of queried ECSs. The value cannot be a negative number. The maximum value is 1,000. <ul style="list-style-type: none">If the action value is count, this parameter is invalid.If the action value is filter, the default value is 1,000.
offset	No	String	Specifies index position. The query starts from the next piece of data indexed by this parameter. The value must be a number and cannot be a negative number. This parameter is not required when data on the first page is queried. When you query the subsequent page data, the value in the response body for the query of the data on the previous page is contained in this parameter. <ul style="list-style-type: none">If the action value is count, this parameter is invalid.If the action value is filter, the default value is 0.
action	Yes	String	Specifies the operation, which can be filter or count . <ul style="list-style-type: none">filter: filters ECSs by tag. The ECSs that meet the filter criteria are displayed on pages.count: searches ECSs by tag. The number of ECSs that meet the search criteria is displayed.

Parameter	Mandatory	Type	Description
matches	No	Array of objects	Specifies the search field, which is used to search for ECSs. Currently, only resource_name can be used for search. For more information, see Table 5-70 .

Table 5-69 tag field description

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. <ul style="list-style-type: none"> A key contains a maximum of 127 Unicode characters. This field cannot be left blank.
values	No	Array of strings	Specifies the tag value. <ul style="list-style-type: none"> Values of the same tag must be unique. Each value contains a maximum of 255 Unicode characters. If this parameter is not specified, any value can be used. The values are in the OR relationship.

Table 5-70 match field description

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the key field to be matched. The tag key can only be resource_name . In such a case, the tag value is the ECS name. <ul style="list-style-type: none"> The key must be unique, and the value is used for matching. This field is a fixed dictionary value. This field cannot be left blank.

Parameter	Mandatory	Type	Description
value	Yes	String	Specifies the tag value. The tag key can only be resource_name . In such a case, the tag value is the ECS name. <ul style="list-style-type: none">Each value contains a maximum of 255 Unicode characters.This field cannot be left blank.

Response

[Table 5-71](#) describes the response parameters.

Table 5-71 Response parameters

Parameter	Type	Description
resources	Array of objects	Specifies returned ECSs. For details, see Table 5-72 .
total_count	Integer	Specifies the total number of queried ECSs.

Table 5-72 resource field description

Parameter	Type	Description
resource_id	String	Specifies the ECS ID.
resource_detail	String	Queries ECS details.
tags	Array of objects	Specifies tags.
resource_name	String	Specifies the resource name, which is the ECS name.

Table 5-73 resource_tag field description

Parameter	Type	Description
key	String	Specifies the tag key. <ul style="list-style-type: none">It contains a maximum of 36 Unicode characters.This field cannot be left blank.Can only consist of digits, letters, hyphens (-), and underscores (_).
value	String	Specifies the tag value. <ul style="list-style-type: none">Each value contains a maximum of 43 Unicode characters.This field can be left blank.Can only consist of digits, letters, hyphens (-), and underscores (_).

Example Request

```
POST https://{endpoint}/v1/{project_id}/servers/resource_instances/action
```

```
{
  "offset": "100",
  "limit": "100",
  "action": "filter",
  "matches": [
    {
      "key": "resource_name",
      "value": "ecs_test"
    }
  ],
  "tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ]
}
```

Example Response

- Response body when **action** is set to **filter**

```
{
  "resources": [
    {
      "resource_detail": null,
      "resource_id": "cdf5_0ef5_wesas_12_dsad",
      "resource_name": "ecs_test",
      "tags": [
        {
          "key": "key1",
          "value": "value1"
        }
      ]
    }
  ],
  "total_count": 1000
}
```


- Response body when **action** is set to **count**

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

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.9.3 Adding Tags to an ECS in a Batch

Function

- This API is used to add tags to a specified ECS in a batch.
- The Tag Management Service (TMS) uses this API to batch manage the tags of an ECS.

Constraints

- An ECS allows a maximum of 10 tags.
- This API is idempotent.
During tag creation, if a tag exists (both the key and value are the same as those of an existing tag), the tag is successfully processed by default.
- A new tag will overwrite the original one if their keys are the same and values are different.

URI

POST /v1/{project_id}/servers/{server_id}/tags/action

[Table 5-74](#) describes the parameters in the URI.

Table 5-74 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-75](#) describes the request parameters.

Table 5-75 Request parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of objects	Specifies the tag list.
action	Yes	String	Specifies the operation. (Only lowercase letters are supported.) For example, create indicates the creation operation.

Table 5-76 resource_tag field description

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. <ul style="list-style-type: none"> Cannot be left blank. Must be unique for each resource. Can contain a maximum of 36 characters. Can be any from Unicode characters (\u4E00-\u9FFF) and the following character set: A-Z, a-z, 0-9, hyphens (-), and underscores (_). The tag key must be unique and cannot be empty.
value	Yes	String	Specifies the tag value. <ul style="list-style-type: none"> Can contain a maximum of 43 characters. Can be any from Unicode characters (\u4E00-\u9FFF) and the following character set: A-Z, a-z, 0-9, hyphens (-), underscores (_), and periods (.).

Response

None

Example Request

POST https://{endpoint}/v1/{project_id}/servers/{server_id}/tags/action

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

```
    "value": "value3"  
  }  
]  
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.9.4 Deleting Tags from an ECS in a Batch

Function

- This API is used to delete tags from a specified ECS in a batch.
- The Tag Management Service (TMS) uses this API to batch manage the tags of an ECS.
- This API is idempotent. When you delete a tag but the tag does not exist, a successful result is returned.

Constraints

An ECS allows a maximum of 10 tags.

URI

POST /v1/{project_id}/servers/{server_id}/tags/action

[Table 5-77](#) describes the parameters in the URI.

Table 5-77 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-78](#) describes the request parameters.

Table 5-78 Request parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of objects	Specifies the tag list.
action	Yes	String	Specifies the operation. (Only lowercase letters are supported.) For example, delete indicates the deletion operation.

Table 5-79 resource_tag field description

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. It contains a maximum of 127 Unicode characters and cannot be left blank. The tag key of an ECS must be unique.
value	No	String	Specifies the tag value. The value can contain a maximum of 255 Unicode characters and can be left blank.

Response

None

Example Request

POST https://{endpoint}/v1/{project_id}/servers/{server_id}/tags/action

```
{
  "action": "delete",
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value3"
    }
  ]
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.9.5 Querying Project Tags

Function

Projects are used to group and isolate OpenStack resources, which include computing, storage, and network resources. A project can be a department or a team. Multiple projects can be created for the same account.

This API is used to query all tags used by a user in a specified project.

URI

GET /v1/{project_id}/servers/tags

[Table 5-80](#) describes the parameters in the URI.

Table 5-80 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

[Table 5-81](#) describes the response parameters.

Table 5-81 Response parameters

Parameter	Type	Description
tags	Array of objects	Specifies tags.

Table 5-82 tag field description

Parameter	Type	Description
key	String	Specifies the tag key. <ul style="list-style-type: none">It contains a maximum of 36 Unicode characters.Can be any from Unicode characters (\u4E00-\u9FFF) and the following character set: A-Z, a-z, 0-9, hyphens (-), and underscores (_).
values	Array of strings	Specifies the tag value. <ul style="list-style-type: none">Each value contains a maximum of 43 Unicode characters.This field can be left blank.Can be any from Unicode characters (\u4E00-\u9FFF) and the following character set: A-Z, a-z, 0-9, hyphens (-), underscores (_), and periods (.).

Examples

- Example request
GET `https://{endpoint}/v1/{project_id}/servers/tags`

- Example response

```
{
  "tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    },
    {
      "key": "key2",
      "values": [
        "value1",
        "value2"
      ]
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.9.6 Querying Tags of an ECS

Function

- This API is used to query the tags of a specified ECS.
- The Tag Management Service (TMS) uses this API to query all tags of an ECS.

URI

GET /v1/{project_id}/servers/{server_id}/tags

[Table 5-83](#) describes the parameters in the URI.

Table 5-83 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-84](#) describes the response parameters.

Table 5-84 Response parameters

Parameter	Type	Description
tags	Array of objects	Specifies the tag list.

Table 5-85 resource_tag field description

Parameter	Type	Description
key	String	Specifies the tag key.
value	String	Specifies the tag value.

Example Request

```
GET https://{endpoint}/v1/{project_id}/servers/{server_id}/tags
```

Example Response

```
{
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value3"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.10 ECS Group Management

5.10.1 Creating an ECS Group

Function

This API is used to create an ECS group.

Compared with the native API for creating an ECS group, this API supports checking fine-grained permissions for enterprise projects.

Constraints

Only anti-affinity groups are supported.

URI

```
POST /v1/{project_id}/cloudservers/os-server-groups
```

[Table 5-86](#) describes the parameters in the URI.

Table 5-86 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 5-87](#) describes the request parameters.

Table 5-87 Request parameters

Parameter	Mandatory	Type	Description
server_group	Yes	Object	Specifies the ECS group information. For details, see Table 5-88 .

Table 5-88 server_group parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the ECS group name. The value contains 1 to 255 characters.
policies	Yes	Array of strings	Specifies the policies associated with the ECS group. Options: <ul style="list-style-type: none">• anti-affinity: ECSs in this group must be deployed on different hosts.

Response

[Table 5-89](#) describes the response parameters.

Table 5-89 Response parameters

Parameter	Type	Description
server_group	Object	Specifies the ECS group information. For details, see Table 5-90 .

Table 5-90 server_group parameters

Parameter	Type	Description
id	String	Specifies the ECS group UUID.
name	String	Specifies the ECS group name.
policies	Array of strings	Specifies the policies associated with the ECS group. Options: <ul style="list-style-type: none">• anti-affinity: ECSs in this group must be deployed on different hosts.
members	Array of strings	Specifies the IDs of the ECSs in an ECS group.
metadata	Object	Specifies the ECS group metadata.

Example Request

Create an ECS group.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/os-server-groups
```

```
{
  "server_group": {
    "name": "test",
    "policies": ["anti-affinity"]
  }
}
```

Example Response

```
{
  "server_group": {
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "name": "test",
    "policies": [
      "anti-affinity"
    ],
    "members": [],
    "metadata": {}
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.10.2 Deleting an ECS Group

Function

This API is used to delete an ECS group.

Compared with the native API for deleting an ECS group, this API supports checking fine-grained permissions for enterprise projects.

URI

DELETE /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}

[Table 5-91](#) describes the parameters in the URI.

Table 5-91 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group UUID.

Request Parameters

None

Response Parameters

None

Example Request

Delete a specified ECS group.

```
DELETE https://{endpoint}/v1/{project_id}/cloudservers/os-server-groups/{server_group_id}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.10.3 Adding an ECS to an ECS Group

Function

This API is used to add an ECS to an ECS group. The system automatically deploys the newly added ECS to a host that is different from the ones accommodating other ECSs in the ECS group.

Constraints

- The ECS to be added has been stopped.
- Only KVM ECSs can be added.
- Only the anti-affinity policy is supported. ECSs in the same ECS group are deployed on different hosts, improving service reliability.
- This API supports checking fine-grained permissions for enterprise projects. For details, see [ecs:cloudServers:create](#).

URI

POST /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}/action

[Table 5-92](#) describes the parameters in the URI.

Table 5-92 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group ID.

Request

[Table 5-93](#) describes the request parameters.

Table 5-93 Request parameters

Parameter	Mandatory	Type	Description
add_member	Yes	Object	Specifies the information of the ECS to be added to an ECS group. For details, see Table 5-94 .

Table 5-94 add_member parameters

Parameter	Mandatory	Type	Description
instance_uuid	Yes	String	Specifies the ECS UUID.

Response

None

Example Request

Adds a specified ECS to an ECS group.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/os-server-groups/{server_group_id}/action
{
  "add_member": {
    "instance_uuid":"34dac9a0-c4a7-457b-bab2-e2c696e0e401"
  }
}
```

Example Response

Status code 200, indicating that the operation is successful

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.10.4 Removing an ECS from an ECS Group

Function

This API is used to remove an ECS from an ECS group. After being removed, the anti-affinity policy will not take effect on this ECS and other ECSs in the same ECS group.

Constraints

Only the anti-affinity policy is supported. ECSs in the same ECS group are deployed on different hosts, improving service reliability.

This API supports checking fine-grained permissions for enterprise projects. For details, see [ecs:cloudServers:delete](#).

URI

POST /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}/action

[Table 5-95](#) describes the parameters in the URI.

Table 5-95 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group ID.

Request

[Table 5-96](#) describes the request parameters.

Table 5-96 Request parameters

Parameter	Mandatory	Type	Description
remove_member	Yes	Object	Specifies the information of the ECS to be removed from an ECS group.

Table 5-97 remove_member parameters

Parameter	Mandatory	Type	Description
instance_uuid	Yes	String	Specifies the ECS UUID.

Response

None

Example Request

Delete the ECS whose UUID is **34dac9a0-c4a7-457b-bab2-e2c696e0e401** from the specified ECS group.

```
POST https://{endpoint}/v1/{project_id}/cloudservers/os-server-groups/{server_group_id}/action
{
  "remove_member": {
    "instance_uuid": "34dac9a0-c4a7-457b-bab2-e2c696e0e401"
  }
}
```

Example Response

Status code 200, indicating that the operation is successful

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.10.5 Querying ECS Groups

Function

This API is used to query ECS groups.

URI

GET /v1/{project_id}/cloudservers/os-server-groups?limit={limit}&marker={marker}

[Table 5-98](#) describes the parameters in the URI.

Table 5-98 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Table 5-99 Query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the upper limit on the number of returned server groups. The maximum value is 1,000.
marker	No	String	Specifies the marker that points to the ECS group. The query starts from the next piece of data indexed by this parameter. Parameters marker and limit must be used together.

Request

None

Response

[Table 5-100](#) describes the response parameters.

Table 5-100 Response parameters

Parameter	Type	Description
server_groups	Array of objects	Specifies ECS groups. For details, see Table 5-101 .
page_info	Object	If the pagination function is enabled, the UUID of the last ECS group on the current page is returned. For details, see Table 5-102 .

Table 5-101 server_groups parameter information

Parameter	Type	Description
id	String	Specifies the ECS group UUID.
name	String	Specifies the ECS group name.
members	Array of strings	Specifies the ECSs contained in an ECS group.
metadata	Object	Specifies the ECS group metadata.
policies	Array of strings	Specifies the policies associated with the ECS group. Options: <ul style="list-style-type: none">• anti-affinity: ECSs in this group must be deployed on different hosts.

Table 5-102 page_info field description

Parameter	Type	Description
next_marker	String	Specifies an ECS group UUID.

Example Request

Query ECS groups.

```
GET https://{endpoint}/v1/{project_id}/cloudservers/os-server-groups
```

Example Response

```
{
  "server_groups": [
    {
```



```
"members": [],
"metadata": {},
"id": "318b44a7-f7a6-4c0b-8107-e8bd618b28dd",
"policies": [
  "anti-affinity"
],
"name": "SvrGrp-b9d6"
},
{
  "members": [],
  "metadata": {},
  "id": "b8f4cfc4-9a59-498c-9b52-643ee6515cd0",
  "policies": [
    "anti-affinity"
  ],
  "name": "SvrGrp-10a1"
}
]
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

5.10.6 Querying Details About an ECS Group

Function

This API is used to query details about an ECS group.

URI

GET /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}

[Table 5-103](#) describes the parameters in the URI.

Table 5-103 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_group_id	Yes	Specifies an ECS group UUID.

Request

None

Response

[Table 5-104](#) describes the response parameters.

Table 5-104 Response parameters

Parameter	Type	Description
server_group	Object	Specifies the ECS group information. For details, see Table 5-105 .

Table 5-105 server_group parameters

Parameter	Type	Description
id	String	Specifies an ECS group UUID.
name	String	Specifies the ECS group name.
policies	Array of strings	Specifies the policies associated with the ECS group.
members	Array of strings	Specifies the ECS contained in an ECS group.
metadata	Object	Specifies the ECS group metadata.

Example Request

Query details about a specified ECS group.

```
GET https://{endpoint}/v1/{project_id}/cloudservers/os-server-groups/{server_group_id}
```

Example Response

```
{
  "server_group": {
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "name": "test",
    "policies": ["anti-affinity"],
    "members": [],
    "metadata": {}
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

6 Native OpenStack Nova APIs

6.1 API Version Query

6.1.1 Querying All API Versions

Function

This API is used to query all available Nova versions.

To support function extension, Nova APIs can be distinguished by version. There are two types of versions:

- Major version: Independent URL
- Microversion: Used by the HTTP request header X-OpenStack-Nova-API-Version. Since microversion 2.27, the new microversion header OpenStack-API-Version has been supported.

URI

GET /

Request

None

Response

The following table describes the response parameters.

Table 6-1 Response parameters

Parameter	Type	Description
versions	Object	Specifies the API versions. For details, see Table 6-2 .

Table 6-2 versions field description

Parameter	Type	Description
id	string	Specifies the version ID.
links	Object	Specifies shortcut links for versions. For details, see Table 6-3 .
min_version	string	<ul style="list-style-type: none">Specifies the microversion. If the APIs of this version support microversions, set this parameter to the supported minimum microversion.If the microversion is not supported, leave this parameter blank.
status	string	Specifies the API version status. Possible values are as follows: <ul style="list-style-type: none">CURRENT: This is the preferred API version.SUPPORTED: This is the old API version that is still supported.DEPRECATED: This is the deprecated API version that will be removed.

Parameter	Type	Description
version	string	<ul style="list-style-type: none">• Specifies the microversion. If the APIs of this version support microversions, set this parameter to the supported maximum microversion.• If the microversion is not supported, leave this parameter blank.
updated	string	The value of this parameter varies by API version. If the API version is 2.0, the value is 2011-01-21T11:33:21Z . If the API version is 2.1, the value is 2013-07-23T11:33:21Z .

Table 6-3 links field description

Parameter	Type	Description
href	string	Specifies the links of the corresponding resources.

Parameter	Type	Description
rel	string	<ul style="list-style-type: none">• self: A self link contains a versioned link to the resource. Use these links when the link is followed immediately.• bookmark: A bookmark link provides a permanent link to a resource that is appropriate for long term storage.• alternate: An alternate link can contain an alternate representation of the resource. For example, an OpenStack Compute image might have an alternate representation in the OpenStack Image service.

Example Request

Query all API versions.

```
GET https://{endpoint}/
```

Example Response

```
{
  "versions": [
    {
      "min_version": "",
      "links": [
        {
          "rel": "self",
          "href": "https://{endpoint}/v2/"
        }
      ]
    },
    {
      "id": "v2.0",
      "updated": "2011-01-21T11:33:21Z",
      "version": "",
      "status": "SUPPORTED"
    }
  ],
  {
    "min_version": "2.1",
    "links": [
      {
        "rel": "self",
        "href": "https://{endpoint}/v2.1/"
      }
    ]
  }
}
```

```
    ],  
    "id": "v2.1",  
    "updated": "2013-07-23T11:33:21Z",  
    "version": "2.60",  
    "status": "CURRENT"  
  }  
]  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.1.2 Querying a Specified API Version

Function Description

This API is used to query the information of a specified version.

To support function extension, Nova APIs can be distinguished by version. There are two types of versions:

- Major version: Independent URL
- Microversion: Used by the HTTP request header X-OpenStack-Nova-API-Version. Since version 2.27, the new microversion header OpenStack-API-Version has been supported.

NOTE

If the OpenStack-API-Version request header is used, the version is in the format of "compute microversion".

For example, if **key** is set to **OpenStack-API-Version**, set **value** to **compute 2.27**.

URI

GET /{api_version}

[Table 6-4](#) describes the parameters in the URI.

Table 6-4 Parameter description

Parameter	Mandatory	Description
api_version	Yes	Specifies an API version, such as V2.

Request

None

Response

The following table describes the response parameters.

Table 6-5 Response parameters

Parameter	Type	Description
versions	Object	Specifies the versions. For details, see Table 6-6 .

Table 6-6 versions field description

Parameter	Type	Description
id	string	Specifies the version ID.
links	Object	Specifies the links to resources. For more information, see the OpenStack Documentation . For details, see Table 6-7 .
media-types	Object	Specifies the media types. For details, see Table 6-8 .
min_version	string	<ul style="list-style-type: none">Specifies the microversion. If the APIs of this version support microversions, set this parameter to the supported minimum microversion.If the microversion is not supported, leave this parameter blank.
status	string	Specifies the API version status. Possible values are as follows: <ul style="list-style-type: none">CURRENT: This is the preferred API version.SUPPORTED: This is the old API version that is still supported.DEPRECATED: This is the deprecated API version that will be removed.

Parameter	Type	Description
updated	string	The value of this parameter varies by API version. If the API version is 2.0, the value is 2011-01-21T11:33:21Z . If the API version is 2.1, the value is 2013-07-23T11:33:21Z .
version	string	<ul style="list-style-type: none">• Specifies the microversion. If the APIs of this version support microversions, set this parameter to the supported maximum microversion.• If the microversion is not supported, leave this parameter blank.

Table 6-7 links field description

Parameter	Type	Description
href	string	Specifies the links of the corresponding resources.

Parameter	Type	Description
rel	string	<ul style="list-style-type: none"> • self: A self link contains a versioned link to the resource. Use these links when the link is followed immediately. • bookmark: A bookmark link provides a permanent link to a resource that is appropriate for long term storage. • alternate: An alternate link can contain an alternate representation of the resource. For example, an OpenStack Compute image might have an alternate representation in the OpenStack Image service.

Table 6-8 media-types field description

Parameter	Type	Description
base	string	Specifies the basic type.
type	string	Specifies the media type.

Example Request

Query information about a specified API version.

```
GET https://{endpoint}/v2.1
```

Example Response

```
{
  "version":{
    "min_version":"2.1",
    "media-types":[
      {
        "type":"application/vnd.openstack.compute+json;version=2.1",
        "base":"application/json"
      }
    ],
    "links":[
```

```
{
  "rel": "self",
  "href": "https://{endpoint}/v2.1/"
},
{
  "rel": "describedby",
  "href": "http://docs.openstack.org/",
  "type": "text/html"
}
],
"id": "v2.1",
"updated": "2013-07-23T11:33:21Z",
"version": "2.60",
"status": "CURRENT"
}
```

Returned Values

See [Returned Values for General Requests](#).

6.2 Lifecycle Management

6.2.1 Creating an ECS

Function

This API is used to create ECS.

This API does not support automatic rollback after creating an ECS failed. If automatic rollback is required, call the API POST `/v1/{project_id}/cloudservers`. For details, see [Creating an ECS](#).

URI

POST `/v2.1/{project_id}/servers`

[Table 6-9](#) describes the parameters in the URI.

Table 6-9 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

NOTE

Alias of the API for creating ECSs: `/v2/{project_id}/os-volumes_boot`
This calling mode can only be used in OpenStack client.


Constraints

1. This API is native, which does not support the creation of ECSs using full-ECS images. To use full-ECS images to create ECSs, refer to [Creating an ECS](#).
2. When using this API to create an ECS, you cannot bind an EIP to the ECS during the creation process. To do so, see [Creating an ECS](#).
3. Parameter **port** in the three network parameters (**port**, **uuid**, and **fixed_ip**) has the highest priority. If parameter **fixed_ip** is set, you must specify the UUID.
4. A file injection failure will result in the ECS creation failure.
5. The following restrictions apply when you create ECSs using an image:
 - a. You cannot create an ECS on a specified host.
 - b. If a tenant backs up a disk in an ECS, the disk can be deleted only after the tenant deletes all the snapshots of the disk.
 - c. The flavors with different resource types cannot be adjusted if you adjust the specifications of an ECS created using an image.
6. Native APIs `/v2/{project_id}/servers` and `/v2.1/{project_id}/servers` provided by the cloud platform is developed based on and compatible with the community-version native OpenStack API.

Compared with the community-version native API, this API has the following restrictions when you create an ECS using a specified image:

- Community-version native OpenStack API: creates an ECS using the local disk by default.
- Native API provided by the cloud platform: creates an ECS using the shared storage as the system disk for reliability.

Specifically, when you use the native API to create an ECS:

- a. You can query information about the system disks attached to the ECS.
 - b. The ECS system disk uses the EVS disk quota.
 - c. You cannot query ECSs created based on a specified image using the image filtering function.
7. When you create an ECS with a specified disk, ensure that the disk and the ECS are in the same AZ.
 8. The **device_name** field configured in **block_device_mapping_v2** during the ECS creation does not take effect. The system generates a device name by default.
 9. ECSs cannot be created in networks with **provider:network_type** set to **geneve**.
-  **NOTE**
- provider:network_type** being set to **geneve** indicates the internal high-speed network for BMSs.
10. If your ECS is remotely logged in using a key, use the **key_name** parameter. If your ECS is remotely logged in using a password, use the **adminPass** parameter. Linux ECSs support **user_data** for injection. Windows ECSs support **admin_pass** for injection.
 11. If the image based on which the ECS is created uses the native OpenStack API, ensure that the specified AZ and system disk capacity and type used

when the ECS is created are the same as those used when the image is created. Otherwise, the ECS creation will fail.

Request

Table 6-10 describes the request parameters.

Table 6-10 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS information. For details, see Table 6-11 .
os:scheduler_hints	No	Object	Specifies the ECS scheduling information. For details, see Table 6-16 . This parameter is not available for BMSs.

Table 6-11 server parameters

Parameter	Mandatory	Type	Description
imageRef	No	String	Specifies the ECS image ID or URL. <ul style="list-style-type: none">• Example image ID: 3b8d6fef-af77-42ab-b8b7-5a7f0af8f2• Example image URL: https://{endpoint}/images/{image_id}• If you use a specified disk as the system disk to create an ECS, this parameter is not required. If you do not use a disk to create an ECS, you must set a valid UUID. Otherwise, the API will return error code 400.
flavorRef	Yes	String	Specifies the flavor ID or URL. For example: c3.2xlarge

Parameter	Mandatory	Type	Description
name	Yes	String	<p>Specifies the ECS name.</p> <p>The value options are as follows:</p> <ul style="list-style-type: none">• The parameter value consists of 1 to 128 characters, including letters, digits, underscores (_), hyphens (-), and periods (.).• If more than one ECS is to be created (the count value is greater than 1), the system automatically adds a hyphen followed by a four-digit incremental number, such as -0000, to the end of each ECS name. If you specify a number, the name of the first new ECS will start from the specified number. In this case, the ECS name contains a maximum of 59 characters. <p>NOTE ECS hostnames comply with RFC952 and RFC1123 naming rules. It is recommended that you configure hostnames using digits, lower-case letters, and hyphens (-).</p>
metadata	No	Map<String, String>	<p>Specifies the ECS metadata. For details, see Table 6-12.</p> <ul style="list-style-type: none">• The key contains 1 to 255 characters.• The value contains 0 to 255 characters.
adminPass	No	String	<p>Specifies the initial login password of the administrator account for logging in to an ECS using password authentication. The Linux administrator is root, and the Windows administrator is Administrator.</p>
block_device_mapping_v2	No	Array of objects	<p>Indicates the V2 API for specifying the ECS storage device. This is an extended attribute. This is the storage resource API of the new version. You are not allowed to create ECSs in batches when the volume is specified. For details, see Table 6-13. This parameter is not available for BMSs.</p>

Parameter	Mandatory	Type	Description
config_drive	No	String	Specifies the config_drive disk to be attached to the ECS during the ECS creation for transferring information to the ECS. This is an extended attribute. This function is not supported.
security_groups	No	Array of objects	Specifies the security group that the ECS belongs to. This parameter is an extended attribute. The default parameter value is default . This parameter is valid when you create an ECS on a specified network. For an existing port, the requested security groups are invalid. For details, see Table 6-14 .
networks	Yes	Array of objects	Specifies information about the ECS NIC. This parameter is an extended attribute. This parameter must be specified if multiple tenant networks are used. For details, see Table 6-15 .
key_name	No	String	Specifies the name of a key pair. This parameter is an extended attribute.

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected to the ECS during the creation. Text and text files can be injected.</p> <p>NOTE</p> <ul style="list-style-type: none"> The content of user_data must be encoded with base64. The maximum size of the content to be injected (before encoding) is 32 KB. <p>For more information about the user data to be injected, see "Injecting User Data into ECSs" in the <i>Elastic Cloud Server User Guide</i>.</p> <p>Examples</p> <p>Before base64 encoding:</p> <ul style="list-style-type: none"> Linux <pre>#!/bin/bash echo user_test > /home/user.txt</pre> Windows <pre>rem cmd echo 111 > c:\aaa.txt</pre> <p>After base64 encoding:</p> <ul style="list-style-type: none"> Linux <pre>lyEvYmluL2Jhc2gKZWNoYmB1c2VvX3Rlc3QgPiAvaG9tZS91c2VvLnR4dA==</pre> Windows <pre>cmVtIGNtZApY2hvIDExMSA+IGM6XGFhYS50eHQ=</pre>
availability_zone	No	String	<p>Specifies the AZ of a specified ECS. This is an extended attribute.</p> <p>You are advised to set this parameter when creating an ECS. Otherwise, the system will select the default AZ.</p>
return_reservation_id	No	Boolean	<p>Specifies whether the reservation IDs of the ECSs created in a batch are returned. This is an extended attribute. You can query the ECSs created this time based on the returned reservation IDs.</p> <ul style="list-style-type: none"> true: The reservation IDs are returned. false: The ECS information is returned. <p>NOTE</p> <p>When you create ECSs in a batch, this parameter is available.</p>

Parameter	Mandatory	Type	Description
min_count	No	Integer	<p>Specifies the minimum number of ECSs that can be created. This is an extended attribute.</p> <p>The default value is 1.</p> <p>NOTE When you use a specified image to create ECSs, this parameter is available.</p>
max_count	No	Integer	<p>Specifies the maximum number of ECSs that can be created.</p> <p>The default value of max_count is the same as that of min_count.</p> <p>Note:</p> <ul style="list-style-type: none"> The max_count value must be greater than or equal to the min_count value. If both min_count and max_count are specified, the number of ECSs that can be created depends on host resources. If host resources permit, you can create a maximum number of ECSs ranging from min_count to max_count values. <p>NOTE When you use a specified image to create ECSs, this parameter is available.</p>
OS-DCF:diskConfig	No	String	<p>Specifies the disk configuration mode. The value can be AUTO or MANUAL.</p> <ul style="list-style-type: none"> MANUAL: indicates that the image space of the system disk cannot be expanded. AUTO: indicates that the image space of the system disk can be automatically expanded to a value same as that specified in flavor. <p>This function is not supported.</p>
description	No	String	<p>Specifies the description of an ECS, which is a null string by default. This is an extended attribute.</p> <p>This parameter is supported in microversion 2.19 and later.</p> <ul style="list-style-type: none"> Can contain a maximum of 85 characters. Cannot contain special characters, such as < and >.

Table 6-12 metadata field description

Parameter	Mandatory	Type	Description
admin_pass	No	String	Specifies the password of user Administrator for logging in to a Windows ECS. NOTE This parameter is mandatory when a Windows ECS using password authentication is created.

Table 6-13 block_device_mapping_v2 parameters

Parameter	Type	Mandatory	Description
source_type	String	Yes	Specifies the source type of the volume device. Its value can be volume , image , snapshot , or blank . If you use a volume to create an ECS, set source_type to volume . If you use an image to create an ECS, set source_type to image . If you use a snapshot to create an ECS, set source_type to snapshot . If you create an empty data volume, set source_type to blank . NOTE If source_type is snapshot and boot_index is 0, the EVS disk of this snapshot must be the system disk.
destination_type	String	No	Specifies the target type of the disk device. Its value can only be volume . <ul style="list-style-type: none">• volume: indicates the volume type.• local: indicates the local file, which has not been supported.
guest_format	String	No	Specifies the local file system format. Its value can be swap or ext4 . This function is not supported.

Parameter	Type	Mandatory	Description
device_name	String	No	Specifies the disk device name. NOTE This field has been discarded. The specified device_name does not take effect. The system generates a device name by default.
delete_on_termination	Boolean	No	Specifies whether disks are deleted when an ECS is deleted. Its default value is false . <ul style="list-style-type: none">• true: When an ECS is deleted, its disks are deleted.• false: When an ECS is deleted, its disks are not deleted.
boot_index	String	No	Specifies whether it is a boot disk. 0 specifies a boot disk, and -1 specifies a non-boot disk. If this parameter is not specified, the default value is -1 . NOTE If source_type of the volume device is volume , there must be one boot_index whose value is 0 .
uuid	String	No	<ul style="list-style-type: none">• If source_type is volume, the value of this parameter is the volume UUID.• If source_type is snapshot, the value of this parameter is the snapshot UUID.• If source_type is image, the value of this parameter is the image UUID.
volume_size	Integer	No	Specifies the volume size. The value is an integer. This parameter is mandatory when source_type is set to image or blank , and destination_type is set to volume . Unit: GB
volume_type	String	No	Specifies the volume type. This parameter is recommended when source_type is set to image and destination_type is set to volume .

Table 6-14 security_groups parameters

Parameter	Mandatory	Type	Description
name	No	String	Specifies the security group name or UUID.

Table 6-15 networks parameters

Parameter	Mandatory	Type	Description
port	No	String	Specifies the network port UUID. This parameter must be set when the network UUID is not specified.
uuid	No	String	Specifies the network UUID. This parameter must be set when the network port is not specified.
fixed_ip	No	String	Specifies the fixed IP address. Parameter port in the three network parameters (port , uuid , and fixed_ip) has the highest priority. If parameter fixed_ip is set, you must specify the UUID.

Table 6-16 os:scheduler_hints parameters

Parameter	Mandatory	Type	Description
group	No	String	Specifies the anti-affinity group. The value is in UUID format. NOTE Ensure that the ECS group uses the anti-affinity policy.
different_host	No	Array of strings	The function has not been supported, and this field is reserved.

Parameter	Mandatory	Type	Description
same_host	No	Array of strings	The function has not been supported, and this field is reserved.
cidr	No	String	The function has not been supported, and this field is reserved.
build_near_host_ip	No	String	The function has not been supported, and this field is reserved.
tenancy	No	String	<p>Specifies whether the ECS is created on a Dedicated Host (DeH) or in a shared pool (default).</p> <p>The value can be shared or dedicated.</p> <ul style="list-style-type: none">• shared: indicates the shared pool.• dedicated: indicates the DeH. <p>The parameter value also takes effect for ECS query operations.</p>
dedicated_host_id	No	String	<p>Specifies the DeH ID.</p> <p>This parameter takes effect only when the value of tenancy is dedicated.</p> <p>If you do not specify this parameter, the system will automatically assign a DeH to you to deploy ECSs.</p> <p>The parameter value also takes effect for ECS query operations.</p>

Response

[Table 6-17](#) describes the response parameters.

Table 6-17 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 6-18 .

Table 6-18 server field description

Parameter	Type	Description
id	String	Specifies the ECS ID in UUID format.
links	Array of objects	Specifies the URI of the ECS. For details, see Table 6-19 .
security_groups	Array of objects	Specifies the security groups to which the ECS belongs. For details, see Table 6-20 .
OS-DCF:diskConfig	String	Specifies the disk configuration mode. <ul style="list-style-type: none">• MANUAL: indicates that the image space of the system disk cannot be expanded.• AUTO: indicates that the image space of the system disk can be automatically expanded to a value same as that specified in flavor.
reservation_id	String	Specifies a filtering criteria to query the created ECSs. NOTE When you create ECSs in a batch, this parameter is available.
adminPass	String	Specifies the password of user Administrator for logging in to a Windows ECS.

Table 6-19 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Provides the corresponding shortcut link.

Table 6-20 security_groups field description

Parameter	Type	Description
name	String	Specifies the security group name or UUID.

Example Request

- Use an image with the extended attribute **block_device_mapping_v2** parameters set to create an ECS and use the key pair for login authentication.
POST <https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers>

```
{
  "server": {
    "flavorRef": "s3.xlarge.2",
    "name": "wjvm48",
    "metadata": {
      "name": "name_xx1",
      "id": "id_xxxx1"
    },
    "block_device_mapping_v2": [{
      "source_type": "image",
      "destination_type": "volume",
      "uuid": "b023fe17-11db-4efb-b800-78882a0e394b",
      "delete_on_termination": "False",
      "boot_index": "0",
      "volume_type": "SAS",
      "volume_size": "40"
    }],
    "security_groups": [{
      "name": "name_xx5_sg"
    }],
    "networks": [{
      "uuid": "fd40e6f8-942d-4b4e-a7ae-465287b02a2c",
      "port": "e730a11c-1a19-49cc-8797-cee2ad67af6f",
      "fixed_ip": "10.20.30.137"
    }],
    "key_name": "test",
    "user_data":
    "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBRbm93IHdoeSBpdCBtb3ZlcyBpbiBqdXN0IHN1Y2ggYSBkaXJl
    Y3Rpb24gYW5kiGF0IHN1Y2ggYSBzcGVlZC4uLkl0IGZlZWxzIGFuIGltcHVsc2lubi4uLnRoXMGaXMgdGhI
    HBsYWNIHRvIGdviG5vdy4gQnV0IHRoZSBza3kga25vd3MgdGhIHRvbnMgYW5kiHRoZSBwYXR0ZXJl
    ucyBiZWhpbmQgYWxslGNsb3VkcycyYW5kiHlvdSB3aWxslGtub3csIHRvbywgd2h1biB5b3UgbGlm
    dCB5b3Vyc2VsZiBoaWd0IGVub3VnaCB0byBzZWUgYmV5b25kiGhvcml6b25zLilNCg0KLVJpY2hhcmQgQmFjaA
    ==",
    "availability_zone": "az1-dc1"
  }
}
```

- Use a snapshot with the extended attribute **block_device_mapping_v2** parameters set to create an ECS. Set **boot_index** to **0** and set the EVS disk corresponding to the snapshot to a system disk.
POST <https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers>

```
{
  "server": {
    "name": "wjvm48",
    "availability_zone": "az1-dc1",
    "block_device_mapping_v2": [
      {
        "source_type": "snapshot",
        "boot_index": "0",
        "uuid": "df51997d-ee35-4fb3-a372-e2ac933a6565", //Specifies the snapshot ID, which is
        "destination_type": "volume"
      }
    ],
    "flavorRef": "s3.xlarge.2",
    "max_count": 1,
    "min_count": 1,
    "networks": [
      {
        "uuid": "79a68cef-0936-4e21-b1f4-b800ecb70246"
      }
    ]
  }
}
```

- Use a volume with the **block_device_mapping_v2** parameters set to create an ECS.

POST <https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers>

```
{
  "server": {
    "flavorRef": "s3.xlarge.2",
    "name": "wjvm48",
    "metadata": {
      "name": "name_xx1",
      "id": "id_xxxx1"
    },
    "block_device_mapping_v2": [{
      "source_type": "volume",
      "destination_type": "volume",
      "uuid": "bd7e4f86-b004-4745-bea2-a55b1085f107",
      "delete_on_termination": "False",
      "boot_index": "0",
      "volume_type": "dsware",
      "volume_size": "40"
    }],
    "security_groups": [{
      "name": "name_xx5_sg"
    }],
    "networks": [{
      "uuid": "fd40e6f8-942d-4b4e-a7ae-465287b02a2c",
      "port": "e730a11c-1a19-49cc-8797-cee2ad67af6f",
      "fixed_ip": "10.20.30.137"
    }],
    "key_name": "test",
    "user_data":
    "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBRbm93IHdoeSBpdCBtb3ZlcyBpbjBqdXN0IHN1Y2ggYSBkaXJl
    Y3Rpb24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLkl0IGZlZWxzIGFuIGltcHVsc2lvbi4uLnRoXMGaXMgdGhll
    HBsYWNIHRvIGdviG5vdy4gQnV0IHRoZSBza3kga25vd3MgdGhllHJlYXNvbnMgYW5kIHRoZSBwYXR0ZXJl
    ucyBiZWVhpbmQgYWxslGNsb3VkcjYwYmV5b25kIGhvdSB3aWxslGtub3csIHRvbywgZ2h1biB5b3UgbGlmdCB5b
    3Vyc2VsZiBoaWd0IGVub3VnaCB0byBzZWUgYmV5b25kIGhvcml6b25zLiINCg0KLVJpY2hhcmQgQmFjaA
    ==",
    "availability_zone": "az1-dc1"
  }
}
```

- Use an imageRef to create an ECS. For security purposes, store the passwords in ciphertext in configuration files or environment variables.

POST <https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers>

```
{
  "server": {
    "flavorRef": "s3.xlarge.2",
    "name": "wjvm48",
    "metadata": {
      "name": "name_xx1",
      "id": "id_xxxx1"
    },
    "adminPass": "$ADMIN_PASS",
    "imageRef": "6b344c54-d606-4e1a-a99e-a7d0250c3d14",
    "security_groups": [{
      "name": "name_xx5_sg"
    }],
    "networks": [{
      "uuid": "fd40e6f8-942d-4b4e-a7ae-465287b02a2c",
      "port": "e730a11c-1a19-49cc-8797-cee2ad67af6f",
      "fixed_ip": "10.20.30.137"
    }],
    "key_name": "test",
    "user_data":
    "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBRbm93IHdoeSBpdCBtb3ZlcyBpbjBqdXN0IHN1Y2ggYSBkaXJl
    Y3Rpb24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLkl0IGZlZWxzIGFuIGltcHVsc2lvbi4uLnRoXMGaXMgdGhll
    HBsYWNIHRvIGdviG5vdy4gQnV0IHRoZSBza3kga25vd3MgdGhllHJlYXNvbnMgYW5kIHRoZSBwYXR0ZXJl
    ucyBiZWVhpbmQgYWxslGNsb3VkcjYwYmV5b25kIGhvdSB3aWxslGtub3csIHRvbywgZ2h1biB5b3UgbGlmdCB5b
    3Vyc2VsZiBoaWd0IGVub3VnaCB0byBzZWUgYmV5b25kIGhvcml6b25zLiINCg0KLVJpY2hhcmQgQmFjaA
    ==",
    "availability_zone": "az1-dc1"
  }
}
```



```
HBsYWNIIHRvIGdviG5vdy4gQnV0IHRoZSBza3kga25vd3MgdGhIIHJlYXNvbnMgYW5kiHRoZSBwYXR0ZXJ  
ncyBiZWVhpbmQgYWxslGNsb3VkcycyYW5kiHlvdSB3aWxslGtub3csIHRvbywgd2hIbI85b3UgbGlmdCB5b  
3Vyc2VsZiBoaWdodGVub3VnaCB0byBzZWUgYmV5b25kiGhvcml6b25zLiINCg0KLVJpY2hhcmQgQmFjaA  
==",  
  "availability_zone": "az1-dc1"  
}  
}
```

- Create at least two and at most three ECSs in a batch.
POST <https://{{endpoint}}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers>

```
{  
  "server": {  
    "availability_zone": "az1.dc1",  
    "name": "test",  
    "imageRef": "10ff4f01-35b6-4209-8397-359cb4475fa0",  
    "flavorRef": "s3.xlarge.2",  
    "return_reservation_id": "true",  
    "networks": [  
      {  
        "uuid": "51bead38-d1a3-4d08-be20-0970c24b7cab"  
      }  
    ],  
    "min_count": "2",  
    "max_count": "3"  
  }  
}
```

Example response

Creating an ECS

```
{  
  "server": {  
    "security_groups": [  
      {  
        "name": "name_xx5_sg"  
      }  
    ],  
    "OS-DCF:diskConfig": "MANUAL",  
    "id": "567c1557-0eca-422c-bfce-149d6b8f1bb8",  
    "links": [  
      {  
        "href": "http://xxx/v2/dc4059e8e7994f2498b514ca04cdaf44/servers/567c1557-0eca-422c-  
bfce-149d6b8f1bb8",  
        "rel": "self"  
      },  
      {  
        "href": "http://xxx/dc4059e8e7994f2498b514ca04cdaf44/servers/567c1557-0eca-422c-  
bfce-149d6b8f1bb8",  
        "rel": "bookmark"  
      }  
    ],  
    "adminPass": "*****"  
  }  
}
```

Creating ECSs in a batch:

```
{  
  "reservation_id": "r-3fhpjulh"  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.2.2 Modifying an ECS

Function

This API is used to modify ECS details. Only the name and description of an ECS can be modified.

URI

PUT /v2.1/{project_id}/servers/{server_id}

[Table 6-21](#) describes the parameters in the URI.

Table 6-21 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-22](#) describes the request parameters.

Table 6-22 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS data structure. For details, see Table 6-23 .

Table 6-23 server field description

Parameter	Mandatory	Type	Description
name	No	String	Specifies the ECS new name. The value is a string of 1 to 255 characters.
description	No	String	Describes the ECS. The value contains a maximum of 255 bytes. This parameter is supported in microversion 2.19 and later.

Response

[Table 6-24](#) describes the response parameters.

Table 6-24 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 6-25 .

Table 6-25 server field description

Parameter	Type	Description
tenant_id	String	Specifies the tenant or project ID.
image	String	Specifies the image ID.
accessIPv4	String	This is a reserved parameter.
addresses	Object	Specifies the attributed network information of the ECS. The structure is Map<String, Object>. For details, see Table 6-26 .
metadata	Object	Specifies the ECS metadata.
accessIPv6	String	This is a reserved parameter.
created	String	Specifies the time when the ECS was created. The time is in the format of "2019-05-22T03:19:19Z".
hostId	String	Specifies the host ID of the ECS.
flavor	Object	Specifies the ECS flavor. For details, see Table 6-27 .
OS-DCF:diskConfig	String	Specifies the disk configuration mode. This is an extended attribute. This field is valid for the ECS started using an image.
user_id	String	Specifies the ID of the user to which an ECS belongs.
name	String	Specifies the modified name of the ECS.
progress	Integer	This is a reserved parameter.
links	Array of Object	Specifies ECS shortcut links. For details, see Table 6-28 .
id	String	Specifies the unique ID of an ECS.

Parameter	Type	Description
updated	String	Specifies the time when the ECS was updated last time. The time is in the format of "2019-05-22T03:19:19Z".
locked	Boolean	Specifies the ECS lock status, which is True when the ECS is locked and False when the ECS is unlocked. This parameter is supported in microversion 2.9 and later.
description	String	Describes the ECS. This parameter is supported in microversion 2.19 and later.
tags	Array of strings	Specifies ECS tags. This parameter is supported in microversion 2.26 and later. If the microversion is not used for query, the response does not contain the tags field. <ul style="list-style-type: none">• The key and value of a tag are connected using an equal sign (=), for example, key=value.• If the value is empty, only the key is returned.
status	String	Specifies the ECS status. Options: ACTIVE, BUILD, ERROR, HARD_REBOOT, MIGRATING, REBOOT, RESIZE, REVERT_RESIZE, SHELVED, SHELVED_OFFLOADED, SHUTOFF, UNKNOWN, and VERIFY_RESIZE For details, see ECS Statuses .

Table 6-26 Data structure of the network which an ECS accesses

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	Specifies the type of an IP address. The value of this parameter can be 4 or 6 . <ul style="list-style-type: none">• 4: The type of the IP address is IPv4.• 6: The type of the IP address is IPv6.

Table 6-27 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID. This parameter is not supported in microversion 2.47 and later.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 6-28 . This parameter is not supported in microversion 2.47 and later.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor. This parameter is supported in microversion 2.47 and later.
ram	Integer	Specifies the memory size (MB) in the ECS flavor. This parameter is supported in microversion 2.47 and later.
disk	Integer	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. This parameter is supported in microversion 2.47 and later.
ephemeral	Integer	Reserved This parameter is supported in microversion 2.47 and later.
swap	Integer	Reserved This parameter is supported in microversion 2.47 and later.
original_name	String	Specifies the name of the ECS flavor. This parameter is supported in microversion 2.47 and later.
extra_specs	Object	Indicates an extended flavor field. For details, see os_extra_specs (flavor) Field Description . This parameter is supported in microversion 2.47 and later.

Table 6-28 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Specifies the shortcut link.

Example Request

Change the name of a specified ECS to **new-server-test**.

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}

{
  "server": {
    "name": "new-server-test"
  }
}
```

Example Response

```
{
  "server": {
    "tenant_id": "7910a6e50b80402ba028c8d96c1b31fe",
    "image": "",
    "accessIPv4": "",
    "addresses": {
      "03be5c1e-e05d-4905-a105-c3bd9b730bdc": [
        {
          "addr": "192.168.0.72",
          "version": 4
        }
      ]
    },
    "metadata": {},
    "accessIPv6": "",
    "created": "2018-05-17T03:15:48Z",
    "hostId": "7dc82f6b1d406200fc63e395cf4829cbffcb49de0e9c75c5773f201f",
    "flavor": {
      "links": [
        {
          "rel": "bookmark",
          "href": "https://None/7910a6e50b80402ba028c8d96c1b31fe/flavors/c3.1U1G"
        }
      ],
      "id": "c3.1U1G"
    },
    "OS-DCF:diskConfig": "MANUAL",
    "user_id": "d698a78532ca430f8daec1858f2b500e",
    "name": "new-server-test",
    "progress": 0,
    "links": [
      {
        "rel": "self",
        "href": "https://None/v2/7910a6e50b80402ba028c8d96c1b31fe/servers/1a19ef4f-be0a-4526-bf2f-14b4464d536a"
      },
      {
        "rel": "bookmark",
        "href": "https://None/7910a6e50b80402ba028c8d96c1b31fe/servers/1a19ef4f-be0a-4526-bf2f-14b4464d536a"
      }
    ],
    "id": "1a19ef4f-be0a-4526-bf2f-14b4464d536a",
    "updated": "2018-05-21T00:36:27Z",
    "status": "ACTIVE"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.2.3 Deleting an ECS

Function

This API is used to delete an ECS.

Constraints

When an ECS is deleted, the NIC that is attached to the ECS and specified by **port_id** through the OpenStack Nova API will be retained, and the NIC specified by **net_id** will be deleted.

URI

DELETE /v2.1/{project_id}/servers/{server_id}

[Table 6-29](#) describes the parameters in the URI.

Table 6-29 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

Delete a specified ECS.

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.2.4 Querying the ECS List

Function

This API is used to query ECSs.

URI

GET /v2.1/{project_id}/servers?changes-since={changes-since}&image={image}&flavor={flavor}&name={name}&status={status}&limit={limit}&marker={marker}¬-tags={not-tags}&reservation_id={reservation_id}&ip={ip}

[Table 6-30](#) describes the parameters in the URI.

Table 6-30 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Table 6-31 Query parameters

Parameter	Mandatory	Type	Description
changes-since	No	String	Specifies the timestamp of the last ECS status update, which is used to filter out the ECSs with statuses updated later than the timestamp. The value is in the format of "CCYY-MM-DDThh:mm:ss+/-hh:mm" in UTC +0 and complies with ISO 8601, for example, 2018-01-17T03:03:32Z.
image	No	String	Specifies the image ID. When image is used as a filter criterion, other filter criteria and paging criteria are not supported. If both the image and other filter criteria are specified, the image filter criterion is used. If the query criteria do not contain the image filter criterion, API functions are not restricted.
flavor	No	String	Specifies the ECS type ID, which is fuzzy-matched.

Parameter	Mandatory	Type	Description
name	No	String	Specifies the ECS name, which is fuzzy-matched.
status	No	String	Specifies the ECS status. Options: ACTIVE, BUILD, ERROR, HARD_REBOOT, MIGRATING, REBOOT, REBUILD, RESIZE, REVERT_RESIZE, SHUTOFF, and VERIFY_RESIZE In microversion 2.37, the system will return an empty list for the status field out of the preceding options. In microversion 2.38 and later, the system will return error 400. For details, see ECS Statuses .
limit	No	Integer	Specifies the upper limit on the number of returned results. The default value on each page is 25, and the information of a maximum of 1,000 ECSs is displayed on each page.
marker	No	String	Specifies the ECS ID to which the marker points. The query will start from its next ID.
tags	No	String	Queries ECSs with tags containing the specified value.
not-tags	No	String	Queries ECSs with tags not containing the specified value.
reservation_id	No	String	Specifies the ID returned when ECSs are created in a batch. This parameter is used to query ECSs created in a batch.
sort_key	No	String	Sorts query results by ECS attribute. The default sorting order is the reverse order of created_at . Options: created_at, availability_zone, display_name, host, instance_type_id, key_name, project_id, user_id, updated_at, uuid, and vm_state
ip	No	String	Indicates the filtering result for IPv4 addresses, which are fuzzy-matched.

Request

None

Response

[Table 6-32](#) describes the response parameters.

Table 6-32 Response parameters

Parameter	Type	Description
servers	Array of objects	Specifies the ECSs to be queried. For details, see Table 6-33 .
servers_links	Array of objects	Specifies the link of the next page in pagination query. For details, see Table 6-34 .

Table 6-33 servers field description

Parameter	Type	Description
name	String	Specifies the ECS name.
id	String	Specifies the unique ID of an ECS.
links	Array of objects	Specifies ECS shortcut links. For details, see Table 6-34 .

Table 6-34 servers_links and links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Specifies the shortcut link.

Example Request

Query a list of ECSs.

```
GET https://{endpoint}/v2.1/{project_id}/servers
```

Example Response

```
{
  "servers": [
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/servers/
```

```
616fb98f-46ca-475e-917e-2563e5a8cd19",
  "rel": "self"
},
{
  "href": "http://openstack.example.com/openstack/servers/
616fb98f-46ca-475e-917e-2563e5a8cd19",
  "rel": "bookmark"
}
],
"name": "new-server-test"
}
]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.2.5 Querying Details About ECSs

Function

This API is used to query details about ECSs.

URI

GET /v2.1/{project_id}/servers/detail?changes-since={changes-since}&image={image}&flavor={flavor}&name={name}&status={status}&limit={limit}&marker={marker}¬-tags={not-tags}&reservation_id={reservation_id}&ip={ip}

[Table 6-35](#) describes the parameters in the URI.

Table 6-35 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Table 6-36 Query parameters

Parameter	Mandatory	Type	Description
changes-since	No	String	Specifies the timestamp of the last ECS status update, which is used to filter out the ECSs with statuses updated later than the timestamp. The format must comply with ISO 8601 in the format of CCYY-MM-DDThh:mm:ss+/-hh:mm, for example, 2018-01-17T03:03:32Z.

Parameter	Mandatory	Type	Description
image	No	String	Specifies the image ID. When image is used as a filter criterion, other filter criteria and paging criteria are not supported. If both the image and other filter criteria are specified, the image filter criterion is used. If the query criteria do not contain the image filter criterion, API functions are not restricted.
flavor	No	String	Specifies the ECS flavor ID, which is fuzzy-matched.
name	No	String	Specifies the ECS name, which is fuzzy-matched.
status	No	String	Specifies the ECS status. Options: ACTIVE, BUILD, ERROR, HARD_REBOOT, MIGRATING, REBOOT, REBUILD, RESIZE, REVERT_RESIZE, SHUTOFF, and VERIFY_RESIZE In microversion 2.37, the system will return an empty list for the status field out of the preceding options. In microversion 2.38 and later, the system will return error 400. For details, see ECS Statuses .
limit	No	Integer	Specifies the upper limit on the number of returned results. Each page contains 25 ECSs by default, and a maximum of 1,000 ECSs are returned. For large data volumes, you are advised to set this parameter to 100 .
marker	No	String	Specifies the ECS ID to which the marker points. The query will start from its next ID.
tags	No	String	Queries ECSs with tags containing the specified value.
not-tags	No	String	Queries ECSs with tags not containing the specified value.

Parameter	Mandatory	Type	Description
reservation_id	No	String	Specifies the ID returned when ECSs are created in a batch. This parameter is used to query ECSs created in a batch.
sort_key	No	String	Sorts query results by ECS attribute. The default sorting order is the reverse order of created_at . The value can be created_at , auto_disk_config , availability_zone , display_description , display_name , host , host_name , image_ref , instance_type_id , kernel_id , key_name , launch_index , launched_at , locked_by , node , power_state , project_id , ramdisk_id , reservation_id , root_device_name , task_state , terminated_at , user_id , updated_at , uuid , or vm_state .
ip	No	String	Indicates the filtering result for IPv4 addresses, which are fuzzy-matched.

Request

None

Response

[Table 6-37](#) describes the response parameters.

Table 6-37 Response parameters

Parameter	Type	Description
servers	Array of objects	Specifies the ECSs to be queried. For details, see Table 6-38 .
servers_links	Array of objects	Specifies the link of the next page in pagination query. For details, see Table 6-40 .

Table 6-38 servers field description

Parameter	Type	Description
name	String	Specifies the ECS name.
id	String	Specifies the unique ID of an ECS.

Parameter	Type	Description
status	String	Specifies the ECS status. Options: ACTIVE, BUILD, DELETED, ERROR, HARD_REBOOT, MIGRATING, PAUSED, REBOOT, REBUILD, RESIZE, REVERT_RESIZE, SHUTOFF, SHELVED, SHELVED_OFFLOADED, SOFT_DELETED, SUSPENDED, and VERIFY_RESIZE For details, see ECS Statuses .
created	String	Specifies the time when the ECS was created. The time is in the format of "2019-05-22T07:48:53Z".
updated	String	Specifies the last time when the ECS was updated, such as started, stopped, or restarted. The time is in the format of "2019-05-22T07:48:53Z".
flavor	Object	Specifies the ECS flavor. For details, see Table 6-39 .
image	Object	Specifies the ECS image information. For an ECS created using an image, the image ID and link are returned. For details, see Table 6-45 .
tenant_id	String	Specifies the ID of the tenant to which the ECS belongs. The parameter value is the same as the project ID specified by project_id .
key_name	String	Specifies the SSH key name.
user_id	String	Specifies the ID of the user to which an ECS belongs.
metadata	Object	Specifies the ECS metadata.
hostId	String	Specifies the host ID of the ECS.
addresses	Object	Specifies the network addresses of an ECS. The structure is Map<String, Object>. <ul style="list-style-type: none">• The key indicates the network name, for example, demo_net.• The value indicates the network attribute specified in Table 6-41.
security_groups	Array of objects	Specifies the security groups to which the ECS belongs. For details, see Table 6-43 .
links	Array of objects	Specifies ECS shortcut links. For details, see Table 6-40 .

Parameter	Type	Description
os:scheduler_hints	Object	Specifies the ECS scheduling information. For details, see Table 6-46 . This parameter is only available for DeHs.
OS-DCF:diskConfig	String	Specifies the disk configuration mode. This is an extended attribute. This field is valid for the ECS started using an image. Options: <ul style="list-style-type: none">• AUTO: This API uses a single partition to build an ECS with the target disk size. The API automatically adjusts the file system to adapt to the entire partition.• MANUAL: This API uses the partitioning scheme in the source image and the file system to build the ECS. If the target disk size is large, the API does not partition the remaining disk space.
OS-EXT-AZ:availability_zone	String	Specifies the AZ ID. This is an extended attribute.
OS-EXT-SRV-ATTR:host	String	Specifies the name of the host on which the ECS is deployed. This is an extended attribute.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the hostname of the hypervisor. This is an extended attribute.
OS-EXT-SRV-ATTR:instance_name	String	Specifies the ECS ID. This is an extended attribute.
OS-EXT-STS:power_state	Integer	Specifies the ECS power status. This is an extended attribute. Options: 0 , 1 , 2 , 3 , and 4 <ul style="list-style-type: none">• 0: pending• 1: running• 2: paused• 3: shutdown• 4: crashed
OS-EXT-STS:task_state	String	Specifies the ECS task status. This is an extended attribute. For details about options, see ECS Statuses .

Parameter	Type	Description
OS-EXT-STS:vm_state	String	Specifies the ECS status. This is an extended attribute. Options: ACTIVE, BUILDING, STOPPED, RESIZED, PAUSED, SUSPENDED, RESCUED, ERROR, DELETED, SOFT_DELETED, SHELVED, and SHELVED_OFFLOADED For details, see ECS Statuses .
OS-SRV-USG:launched_at	String	Specifies the time when the ECS was started. This is an extended attribute. The time is in the format of "2019-05-22T07:48:19.000000".
OS-SRV-USG:terminated_at	String	Specifies the time when the ECS was deleted. This is an extended attribute. The time is in the format of "2019-05-22T07:48:19.000000".
os-extended-volumes:volumes_attached	Array of objects	Specifies information about the EVS disks attached to the ECS. For details, see Table 6-42 .
fault	Object	Describes ECS faults. This parameter is optional. It is returned when an error occurs on an ECS. For details, see Table 6-44 .
description	String	Describes the ECS. This parameter is supported in microversion 2.19 and later.
host_status	String	Specifies the nova-compute status. <ul style="list-style-type: none">● UP: The nova-compute status is normal.● UNKNOWN: The nova-compute status is unknown.● DOWN: the nova-compute status is abnormal.● MAINTENANCE: The nova-compute is in maintenance state.● Empty string: There is no host information on the ECS. This parameter is supported in microversion 2.16 and later.
OS-EXT-SRV-ATTR:hostname	String	Specifies the name of the host accommodating the ECS. This parameter is supported in microversion 2.3 and later.

Parameter	Type	Description
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the reserved ECS ID if multiple ECSs are created in a batch. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the sequence in which ECSs created in a batch start. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:kernel_id	String	Specifies the UUID of the kernel image if an AMI image is used. In other scenarios, leave this parameter blank. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:ramdisk_id	String	Specifies the UUID of the ramdisk image if an AMI image is used. In other scenarios, leave this parameter blank. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the ECS system disk. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:user_data	String	Specifies the user data specified during ECS creation. This parameter is supported in microversion 2.3 and later.
tags	Array of strings	Specifies ECS tags. This parameter is supported in microversion 2.26 and later. If the microversion is not used for query, the response does not contain the tags field.
locked	Boolean	Specifies the ECS lock status, which is True when the ECS is locked and False when the ECS is unlocked. This parameter is supported in microversion 2.9 and later.
accessIPv4	String	This is a reserved parameter.
accessIPv6	String	This is a reserved parameter.
config_drive	String	This is a reserved parameter.
progress	Integer	This is a reserved parameter.

Table 6-39 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID. This parameter is not supported in microversion 2.47 and later.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 6-40 . This parameter is not supported in microversion 2.47 and later.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor. This parameter is supported in microversion 2.47 and later.
ram	Integer	Specifies the memory size (MB) in the ECS flavor. This parameter is supported in microversion 2.47 and later.
disk	Integer	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. This parameter is supported in microversion 2.47 and later.
ephemeral	Integer	Reserved This parameter is supported in microversion 2.47 and later.
swap	Integer	Reserved This parameter is supported in microversion 2.47 and later.
original_name	String	Specifies the name of the ECS flavor. This parameter is supported in microversion 2.47 and later.
extra_specs	Object	Extended flavor field For details, see Data Structure for Querying Details About Specifications . This parameter is supported in microversion 2.47 and later.

Table 6-40 servers_links and links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Specifies the shortcut link.

Table 6-41 Data structure of the network which an ECS accesses

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	Specifies the type of an IP address. The value of this parameter can be 4 or 6 . <ul style="list-style-type: none">• 4: The type of the IP address is IPv4.• 6: The type of the IP address is IPv6.
OS-EXT-IPS-MAC:mac_address	String	Specifies the MAC address. This is an extended attribute.
OS-EXT-IPS:type	String	Specifies the IP address assignment mode. This is an extended attribute.

Table 6-42 os-extended-volumes:volumes_attached field description

Parameter	Type	Description
id	String	Specifies the EVS disk ID.
delete_on_termination	Boolean	Specifies whether to delete additional disks when deleting the ECS. By default, this parameter is set to False . This parameter is supported in microversion 2.3 and later.

Table 6-43 security_groups field description

Parameter	Type	Description
name	String	Specifies the security group name or UUID.

Table 6-44 fault field description

Parameter	Type	Description
code	Integer	Specifies the error code.
created	String	Specifies the time when an error occurred.
message	String	Describes an error.
details	String	Specifies details about an error. This parameter is optional and is returned only when it is not empty.

Table 6-45 image field description

Parameter	Type	Description
id	String	Specifies the image ID.
links	Array of objects	Specifies shortcut links for ECS images. For details, see Table 6-40 .

Table 6-46 os:scheduler_hints parameters

Parameter	Mandatory	Type	Description
tenancy	No	Array of strings	Creates ECSs on a dedicated or shared host. The value of this parameter can be dedicated or shared .
dedicated_host_id	No	Array of strings	Specifies the DeH ID. This parameter takes effect only when tenancy is set to dedicated .

Example Request

Query details about ECSs.

```
GET https://{endpoint}/v2.1/{project_id}/servers/detail
```

Example Response

```
{
  "servers": [
    {
      "addresses": {
        "68269e6e-4a27-441b-8029-35373ad50bd9": [
          {
            "addr": "192.168.0.3",
            "version": 4
          }
        ]
      },
      "created": "2012-09-07T16:56:37Z",
      "flavor": {
        "id": "s3.xlarge.2",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/flavors/s3.xlarge.2",
            "rel": "bookmark"
          }
        ]
      }
    },
    {
      "hostId": "16d193736a5cfdb60c697ca27ad071d6126fa13baeb670fc9d10645e",
      "id": "05184ba3-00ba-4fbc-b7a2-03b62b884931",
      "image": "",
      "links": [
```

```
{
  {
    "href": "http://openstack.example.com/v2/openstack/servers/05184ba3-00ba-4fbc-
b7a2-03b62b884931",
    "rel": "self"
  },
  {
    "href": "http://openstack.example.com/openstack/servers/05184ba3-00ba-4fbc-
b7a2-03b62b884931",
    "rel": "bookmark"
  }
],
"metadata": {},
"name": "new-server-test",
"progress": 0,
"status": "ACTIVE",
"tenant_id": "openstack",
"updated": "2012-09-07T16:56:37Z",
"user_id": "fake"
}
]
```

Returned Values

See [Returned Values for General Requests](#).

6.2.6 Querying Details About an ECS

Function

This API is used to query details about an ECS by ECS ID.

URI

GET /v2.1/{project_id}/servers/{server_id}

[Table 6-47](#) describes the parameters in the URI.

Table 6-47 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-48](#) describes the response parameters.

Table 6-48 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 6-49 .

Table 6-49 server field description

Parameter	Type	Description
name	String	Specifies the ECS name.
id	String	Specifies the unique ID of an ECS.
status	String	Specifies the ECS status. Options: ACTIVE, BUILD, DELETED, ERROR, HARD_REBOOT, MIGRATING, PAUSED, REBOOT, REBUILD, RESIZE, REVERT_RESIZE, SHUTOFF, SHELVED, SHELVED_OFFLOADED, SOFT_DELETED, SUSPENDED, and VERIFY_RESIZE For details, see ECS Statuses .
created	String	Specifies the time when the ECS was created. The time is in the format of "2019-05-22T07:48:19Z".
updated	String	Specifies the last time when the ECS was updated, such as started, stopped, or restarted. The time is in the format of "2019-05-22T07:48:19Z".
flavor	Object	Specifies the ECS flavor. For details, see Table 6-50 .
image	Object	Specifies the ECS image information. For an ECS created using an image, the image ID and link are returned. For details, see Table 6-51 .
tenant_id	String	Specifies the ID of the tenant to which the ECS belongs. The parameter value is the same as the project ID specified by project_id .
key_name	String	Specifies the SSH key name.
user_id	String	Specifies the ID of the user to which an ECS belongs.
metadata	Object	Specifies the ECS metadata.
hostId	String	Specifies the host ID of the ECS.

Parameter	Type	Description
addresses	Object	Specifies the network addresses of an ECS. The structure is Map<String, Object>. <ul style="list-style-type: none"> The key indicates the network name, for example, demo_net. The value indicates the network attribute specified in Table 6-53.
security_group_s	Array of objects	Specifies the security groups to which the ECS belongs. For details, see Table 6-55 .
links	Array of objects	Specifies ECS shortcut links. For details, see Table 6-52 .
tags	Array of strings	Specifies ECS tags. This parameter is supported in microversion 2.26 and later. If the microversion is not used for query, the response does not contain the tags field.
os:scheduler_hints	Object	Specifies the ECS scheduling information. For details, see Table 6-57 . This parameter is not available for BMSs. and is only available in DeH scenarios.
OS-DCF:diskConfig	String	Specifies the disk configuration mode. This is an extended attribute. This field is valid for the ECS started using an image. <p>Options:</p> <ul style="list-style-type: none"> AUTO: This API uses a single partition to build an ECS with the target disk size. The API automatically adjusts the file system to adapt to the entire partition. MANUAL: This API uses the partitioning scheme in the source image and the file system to build the ECS. If the target disk size is large, the API does not partition the remaining disk space.
OS-EXT-AZ:availability_zone	String	Specifies the AZ ID. This is an extended attribute.
OS-EXT-SRV-ATTR:host	String	Specifies the name of the host on which the ECS is deployed. This is an extended attribute.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the hostname of the hypervisor. This is an extended attribute.
OS-EXT-SRV-ATTR:instance_name	String	Specifies the ECS ID. This is an extended attribute.

Parameter	Type	Description
OS-EXT-STS:power_state	Integer	Specifies the ECS power status. This is an extended attribute. Options: 0, 1, 2, 3, and 4 <ul style="list-style-type: none">• 0: pending• 1: running• 2: paused• 3: shutdown• 4: crashed
OS-EXT-STS:task_state	String	Specifies the ECS task status. This is an extended attribute. For details about options, see ECS Statuses .
OS-EXT-STS:vm_state	String	Specifies the ECS status. This is an extended attribute. Options: ACTIVE, BUILDING, STOPPED, RESIZED, PAUSED, SUSPENDED, RESCUED, ERROR, DELETED, SOFT_DELETED, SHELVED, and SHELVED_OFFLOADED For details, see ECS Statuses .
OS-SRV-USG:launched_at	String	Specifies the time when the ECS was started. This is an extended attribute. The time is in the format of "2019-05-22T07:48:19.000000".
OS-SRV-USG:terminated_at	String	Specifies the time when the ECS was deleted. This is an extended attribute. The time is in the format of "2019-05-22T07:48:19.000000".
os-extended-volumes:volumes_attached	Array of objects	Specifies information about the EVS disks attached to the ECS. For details, see Table 6-54 .
fault	Object	Describes ECS faults. This parameter is optional. It is returned when an error occurs on an ECS. For details, see Table 6-56 .
description	String	Describes the ECS. This parameter is supported in microversion 2.19 and later.

Parameter	Type	Description
host_status	String	Specifies the nova-compute status. <ul style="list-style-type: none">• UP: The nova-compute status is normal.• UNKNOWN: The nova-compute status is unknown.• DOWN: the nova-compute status is abnormal.• MAINTENANCE: The nova-compute is in maintenance state.• Null: There is no host information on the ECS. This parameter is supported in microversion 2.16 and later.
OS-EXT-SRV-ATTR:hostname	String	Specifies the name of the host accommodating the ECS. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the reserved ECS ID if multiple ECSs are created in a batch. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the sequence in which ECSs created in a batch start. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:kernel_id	String	Specifies the UUID of the kernel image if an AMI image is used. In other scenarios, leave this parameter blank. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:ramdisk_id	String	Specifies the UUID of the ramdisk image if an AMI image is used. In other scenarios, leave this parameter blank. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the ECS system disk. This parameter is supported in microversion 2.3 and later.
OS-EXT-SRV-ATTR:user_data	String	Specifies the user data specified during ECS creation. This parameter is supported in microversion 2.3 and later.

Parameter	Type	Description
locked	Boolean	Specifies the ECS lock status, which is True when the ECS is locked and False when the ECS is unlocked. This parameter is supported in microversion 2.9 and later.
accessIPv4	String	This is a reserved parameter.
accessIPv6	String	This is a reserved parameter.
config_drive	String	This is a reserved parameter.
progress	Integer	This is a reserved parameter.

Table 6-50 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID. This parameter is not supported in microversion 2.47 and later.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 6-52 . This parameter is not supported in microversion 2.47 and later.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor. This parameter is supported in microversion 2.47 and later.
ram	Integer	Specifies the memory size (MB) in the ECS flavor. This parameter is supported in microversion 2.47 and later.
disk	Integer	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. This parameter is supported in microversion 2.47 and later.
ephemeral	Integer	Reserved This parameter is supported in microversion 2.47 and later.
swap	Integer	Reserved This parameter is supported in microversion 2.47 and later.

Parameter	Type	Description
original_name	String	Specifies the name of the ECS flavor. This parameter is supported in microversion 2.47 and later.
extra_specs	Object	Indicates an extended flavor field. For details, see os_extra_specs (flavor) Field Description . This parameter is supported in microversion 2.47 and later.

Table 6-51 image field description

Parameter	Type	Description
id	String	Specifies the image ID.
links	Array of objects	Specifies shortcut links for ECS images. For details, see Table 6-52 .

Table 6-52 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Provides the corresponding shortcut link.

Table 6-53 Data structure of the network which an ECS accesses

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	Specifies the type of an IP address. The value of this parameter can be 4 or 6 . <ul style="list-style-type: none">• 4: The type of the IP address is IPv4.• 6: The type of the IP address is IPv6.
OS-EXT-IPS-MAC:mac_address	String	Specifies the MAC address. This is an extended attribute.
OS-EXT-IPS:type	String	Specifies the IP address assignment mode. This is an extended attribute.

Table 6-54 os-extended-volumes:volumes_attached field description

Parameter	Type	Description
id	String	Specifies the EVS disk ID.
delete_on_termination	Boolean	Specifies whether to delete additional disks when deleting the ECS. By default, this parameter is set to False . This parameter is supported in microversion 2.3 and later.

Table 6-55 security_groups field description

Parameter	Type	Description
name	String	Specifies the security group name or UUID.

Table 6-56 fault field description

Parameter	Type	Description
code	Integer	Specifies the error code.
created	String	Specifies the time when an error occurred.
message	String	Describes an error.
details	String	Specifies details about an error. This parameter is optional and is returned only when it is not empty.

Table 6-57 os:scheduler_hints parameters

Parameter	Mandatory	Type	Description
tenancy	No	Array of strings	Creates ECSs on a dedicated or shared host. The value of this parameter can be dedicated or shared .
dedicated_host_id	No	Array of strings	Specifies the DeH ID. This parameter takes effect only when the value of tenancy is dedicated .

Example Request

Query details about a specified ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}
```

Example Response

```
{
  "server": {
    "addresses": {
      "68269e6e-4a27-441b-8029-35373ad50bd9": [
        {
          "addr": "192.168.0.3",
          "version": 4,
          "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:1b:35:78",
          "OS-EXT-IPS:type": "fixed"
        }
      ]
    },
    "created": "2012-08-20T21:11:09Z",
    "flavor": {
      "id": "s3.xlarge.2",
      "links": [
        {
          "href": "http://openstack.example.com/openstack/flavors/s3.xlarge.2",
          "rel": "bookmark"
        }
      ]
    },
    "hostId": "65201c14a29663e06d0748e561207d998b343e1d164bfa0aafa9c45d",
    "id": "893c7791-f1df-4c3d-8383-3caae9656c62",
    "image": "",
    "links": [
      {
        "href": "http://openstack.example.com/v2/openstack/servers/893c7791-f1df-4c3d-8383-3caae9656c62",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/openstack/servers/893c7791-f1df-4c3d-8383-3caae9656c62",
        "rel": "bookmark"
      }
    ],
    "metadata": {},
    "name": "new-server-test",
    "progress": 0,
    "status": "ACTIVE",
    "tenant_id": "openstack",
    "updated": "2012-08-20T21:11:09Z",
    "user_id": "fake"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.3 Status Management

6.3.1 Starting an ECS

Function

This API is used to start a single ECS.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-58](#) describes the parameters in the URI.

Table 6-58 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-59](#) describes the request parameters.

Table 6-59 Request parameters

Parameter	Mandatory	Type	Description
os-start	Yes	Null	Specifies the operation to start the ECS. The data structure is empty.

Response

None

Example Request

Start a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "os-start": {}
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.2 Restarting an ECS

Function

This API is used to restart a single ECS.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-60](#) describes the parameters in the URI.

Table 6-60 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-61](#) describes the request parameters.

Table 6-61 Request parameters

Parameter	Mandatory	Type	Description
reboot	Yes	Object	Specifies the operation to restart the ECS. For details, see Table 6-62 .

Table 6-62 reboot field description

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the type of the restart operation. <ul style="list-style-type: none">• SOFT: soft restart• HARD: forcible restart (hard restart)

Response

None

Example Request

Restart a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "reboot": {
    "type": "SOFT"
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.3 Disabling an ECS

Function

This API is used to stop a single ECS.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-63](#) describes the parameters in the URI.

Table 6-63 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-64](#) describes the request parameters.

Table 6-64 Request parameters

Parameter	Mandatory	Type	Description
os-stop	Yes	Object	Specifies the operation to stop the ECS. For details, see Table 6-65 .

Table 6-65 os-stop field description

Parameter	Mandatory	Type	Description
type	No	String	Specifies an ECS stop type. The default value is SOFT . <ul style="list-style-type: none">• SOFT: normal ECS stop• HARD: forcible ECS stop

Response

None

Example Request

Stop a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "os-stop": {}
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.4 Locking an ECS

Function

This API is used to lock an ECS.

You are only allowed to lock your own ECSs. After ECSs are locked, you will not be able to perform management operations on them, including life cycle management, status management, NIC management, disk management, and password management.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-66](#) describes the parameters in the URI.

Table 6-66 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-67](#) describes the request parameters.

Table 6-67 Request parameters

Parameter	Type	Mandatory	Description
lock	Null	Yes	Locks an ECS.

Response

None

Example Request

Lock a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "lock": null
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.5 Unlocking an ECS

Function

This API is used to unlock an ECS.

After an ECS is unlocked, common users are allowed to manage the ECS.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-68](#) describes the parameters in the URI.

Table 6-68 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-69](#) describes the request parameters.

Table 6-69 Request parameters

Parameter	Mandatory	Type	Description
unlock	Yes	Null	Unlocks an ECS.

Response

None

Example Request

Unlock a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "unlock": null
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.6 Rebuilding an ECS

Function

This API is used to rebuild an ECS.

You can use the original image or another image to rebuild an ECS. This API supports different OSs.

This API is native from the community for defcore tests.

If you are required to reinstall or change an ECS OS, ECS APIs are recommended. For details, see [Reinstalling an ECS OS \(Using an Image with Cloud-Init Installed\)](#) and [Changing an ECS OS \(Using an Image with Cloud-Init Installed\)](#).

Constraints

- ECSs in the error state cannot be rebuilt.
- The password cannot be set during the rebuilding.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-70](#) describes the parameters in the URI.

Table 6-70 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-71](#) describes the request parameters.

Table 6-71 Request parameter

Parameter	Mandatory	Type	Description
rebuild	Yes	Object	Rebuilds an ECS. For details, see Table 6-72 .

Table 6-72 rebuild parameters

Parameter	Mandatory	Type	Description
name	No	String	Specifies the name of the rebuilt ECS. The value contains 1 to 254 characters.
metadata	No	Object	Specifies the metadata of the rebuilt ECS.
imageRef	Yes	String	Specifies the image ID or URL.
adminPass	No	String	Specifies the password for logging in to the rebuilt ECS. This parameter does not take effect.
OS-DCF:diskConfig	No	String	Not supported
preserve_ephemeral	No	Boolean	Specifies whether to retain the temporary disk. This parameter is not supported.

Response

[Table 6-73](#) describes the response parameters.

Table 6-73 Response parameters

Parameter	Type	Description
status	String	Specifies the ECS status. Values: <ul style="list-style-type: none">• ACTIVE• REBOOT• HARD_REBOOT• REBUILD• MIGRATING• BUILD• SHUTOFF• RESIZE• VERIFY_RESIZE• ERROR• DELETED For details, see ECS Statuses .
updated	String	Specifies the time when the ECS was updated last time.
hostId	String	Specifies the ID of the host on which the ECS is deployed.
addresses	Array of objects	Specifies the network attribute of the ECS.
links	Array of objects	Describes the ECS.
image	Object	Specifies the ECS image information. For the ECS that boots from a volume, the value is left blank.
flavor	Object	Specifies the ECS flavor.
id	String	Specifies the ECS ID in UUID format.
user_id	String	Specifies the user UUID of the ECS.
name	String	Specifies the ECS name.
tenant_id	String	Specifies the UUID of the tenant who owns the ECS.

Parameter	Type	Description
OS-DCF:diskConfig	String	Specifies the diskConfig type. It is an extended attributed. <ul style="list-style-type: none">• MANUAL: The image space cannot be expanded.• AUTO: The image space on the system disk will be automatically expanded to keep the same as that set in the flavor.
accessIPv4	String	Discarded
accessIPv6	String	Discarded
progress	String	Specifies the ECS creation progress.
metadata	Object	Specifies the ECS metadata.

Example Request

Use the **3ed456f5-3d8f-4383-a6c9-312032afcd1a** image to rebuild an ECS.

POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action

```
{
  "rebuild" : {
    "imageRef" : "3ed456f5-3d8f-4383-a6c9-312032afcd1a",
    "name" : "rebuildName",
    "metadata" : {
      "rebuild" : "rebuild vm"
    }
  }
}
```

Example Response

```
{
  "server": {
    "tenant_id": "7459f9935ed2422eb9800fea1d4d9378",
    "image": {
      "links": [
        {
          "rel": "bookmark",
          "href": "https://xxx/7459f9935ed2422eb9800fea1d4d9378/images/3ed456f5-3d8f-4383-a6c9-312032afcd1a"
        }
      ],
      "id": "3ed456f5-3d8f-4383-a6c9-312032afcd1a"
    },
    "accessIPv4": "",
    "addresses": {
      "443dd9e3-c165-4764-ad92-b17fcf12a3eb": [
        {
          "addr": "192.168.0.119",
          "version": 4
        }
      ]
    },
    "metadata": {
      "name": "rebuildName"
    }
  }
}
```

```
    },
    "accessIPv6": "",
    "created": "2016-09-19T01:13:26Z",
    "hostId": "fd16ebd9c2629e8595875cc1e1400fa67f392431d7937fcc9cf37671",
    "adminPass": "qGVjnEjY3ZoY",
    "flavor": {
      "links": [
        {
          "rel": "bookmark",
          "href": "https://xxx/7459f9935ed2422eb9800fea1d4d9378/flavors/s3.xlarge.1"
        }
      ]
    },
    "id": "s3.xlarge.1"
  },
  "OS-DCF:diskConfig": "MANUAL",
  "user_id": "ed2965d80d394be0b41e56f50ac650ca",
  "name": "rebuildName",
  "progress": 0,
  "links": [
    {
      "rel": "self",
      "href": "https://xxx/v2/7459f9935ed2422eb9800fea1d4d9378/servers/ea681a24-9b24-4f49-98ef-8e1f73acf19e"
    },
    {
      "rel": "bookmark",
      "href": "https://xxx/7459f9935ed2422eb9800fea1d4d9378/servers/ea681a24-9b24-4f49-98ef-8e1f73acf19e"
    }
  ],
  "id": "ea681a24-9b24-4f49-98ef-8e1f73acf19e",
  "updated": "2016-09-19T07:22:05Z",
  "status": "REBUILD"
}
```

Returned Values

See [Returned Values for General Requests](#).

6.3.7 Creating an Image Using an ECS

Function

This API is used to create an image using an ECS. After the creation, you can use this image to create ECSs.

Images created using an ECS are stored on storage nodes as snapshots.

NOTE

This API is a native OpenStack API that is not applicable to the images on the cloud platform.

- To create a system disk image or data disk image, use the IMS API (**POST /v2/cloudimages/action**). For details, see "Creating an Image" in *Image Management Service API Reference*.
- To create a full-ECS image, use the IMS API (**POST /v1/cloudimages/wholeimages/action**). For details, see "Creating a Full-ECS Image" in *Image Management Service API Reference*.

Constraints

1. An ECS in the error state cannot be used to create an image.
2. If an image created using an ECS is used to create a new ECS, the new ECS must be located in the same AZ as the original ECS.
3. After an image created using an ECS is deleted, the associated snapshots will not be automatically deleted (this function is implemented by native OpenStack). You must manually delete such snapshots.
4. The image created using an ECS cannot be used to create data disks.
5. The images created using the API described in this section (URI: POST /v2/{project_id}/servers/{server_id}/action or POST /v2.1/{project_id}/servers/{server_id}/action) cannot be exported to OBS buckets. If such images must be exported, use the IMS API (**POST /v2/cloudimages/action**). For details, see "Creating an Image" in *Image Management Service API Reference*.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-74](#) describes the parameters in the URI.

Table 6-74 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-75](#) describes the request parameters.

Table 6-75 Request parameters

Parameter	Mandatory	Type	Description
createImage	Yes	Object	Specifies the image created using ECS. For details, see Table 6-76 .

Table 6-76 createImage field description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the image name with a length greater than 0 bytes and less than 243 bytes.
metadata	No	Object	Specifies the image attribute with a length greater than 0 bytes and less than 255 bytes.

Response

Parameter	Mandatory	Type	Description
Location	Yes	String	Specifies the local URL of the image, which is returned in the request header. This parameter is not supported in microversion 2.44 and later.
image_id	Yes	String	Specifies the image UUID. This parameter is supported in microversion 2.45 and later.

Example Request

Use a specified ECS to create a private image named **new-image-name**.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
```

```
{
  "createImage": {
    "name": "new-image-name",
    "metadata": {
      "ImageType": "Gold",
      "ImageVersion": "2.0"
    }
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.8 Modifying the Specifications of an ECS

Function

This API is used to modify the specifications of an ECS.

For a running ECS, the system will automatically stop the ECS, copy the ECS data to the target node, which can be the source node, and then restart the ECS.

This API supports automatic rollback if the underlying resources are insufficient.

This API must be used with the API for verifying ECS specifications modification (POST /v2.1/{project_id}/servers/{server_id}/action) or the API for rolling back ECS specifications modification (POST /v2.1/{project_id}/servers/{server_id}/action) if an ECS is detected to be in **VERIFY_RESIZE** state and its **OS-EXT-STS:vm_state** is **RESIZED**.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-77](#) describes the parameters in the URI.

Table 6-77 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-78](#) describes the request parameters.

Table 6-78 Request parameters

Parameter	Mandatory	Type	Description
resize	Yes	Object	For details about how to modify specifications, see Table 6-79 .

Table 6-79 resize field description

Parameter	Mandatory	Type	Description
flavorRef	Yes	String	Specifies the new flavor ID or URI.

Response

None

Example Request

Change the flavor of a specified ECS to **s3.medium.2**.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "resize": {
    "flavorRef": "s3.medium.2"
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.9 Confirming the Specifications Modification of an ECS

Function

This API is used to confirm the specifications modification of an ECS.

Constraints

Before calling this API, ensure that the ECS status (which can be queried using the API for querying details about the ECS) meets the following requirements:

OS-EXT-STS:vm_state=resized

OS-EXT-STS:task_state=""

status=VERIFY_RESIZE

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-80](#) describes the parameters in the URI.

Table 6-80 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-81](#) describes the request parameters.

Table 6-81 Request parameters

Parameter	Mandatory	Type	Description
confirmResize	Yes	Null	Confirms the modification to ECS specifications.

Response

None

Example Request

Confirm the modifications to the specifications of a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "confirmResize" : null
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.10 Rolling Back ECS Specifications Modification

Function

This API is used to roll back ECS specifications modification.

Constraints

After the rollback, the data modified during migration will be lost.

Before calling this API, ensure that the ECS status (which can be queried using the API for querying details about the ECS) meets the following requirements:

OS-EXT-STS:vm_state=resized

OS-EXT-STS:task_state=""

status=VERIFY_RESIZE

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-82](#) describes the parameters in the URI.

Table 6-82 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-83](#) describes the request parameters.

Table 6-83 Request parameters

Parameter	Mandatory	Type	Description
revertResize	Yes	Null	Confirms the rollback of the ECS specification modification.

Response

None

Example Request

Roll back modifications to the specifications of a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "revertResize" : null
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.3.11 Adding an ECS to the Monitoring List

Function

This API is used to add an ECS to the monitoring list.

Ceilometer periodically collects monitoring data on the ECSs added to the monitoring list and reports the data to Cloud Eye. The data includes the platform version, CPU, memory, NICs, disks, and hardware version. For example, the plug-in of an SAP ECS periodically obtains monitoring data from Cloud Eye and reports the data to SAP in reports.

URI

POST /v1.0/servers/{server_id}/action

[Table 6-84](#) describes the parameters in the URI.

Table 6-84 Parameter description

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-85](#) describes the request parameters.

Table 6-85 Request parameters

Parameter	Mandatory	Type	Description
monitorMetrics	Yes	Null	Adds an ECS to the monitoring list.

Response

None

Example Request

Add a specified ECS to the monitoring list.

```
POST https://{endpoint}/v1.0/servers/{server_id}/action
{
  "monitorMetrics" : null
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

6.4 Network Management

6.4.1 Querying Networks

Function

This API is used to query the networks available to a tenant.

Constraints

You can query only the network ID and label (network name). Other fields are all null.

URI

GET /v2.1/{project_id}/os-networks

[Table 6-86](#) describes the parameters in the URI.

Table 6-86 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

Table 6-87 Parameter description

Parameter	Mandatory	Type	Description
networks	Yes	Array of objects	Specifies the network where the ECS accesses. For details, see Table 6-88 .

Table 6-88 Response parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the network ID in UUID format.
label	Yes	String	Specifies the network name.
broadcast	Yes	String	The value can only be null.
cidr	Yes	String	The value can only be null.
cidr_v6	Yes	String	The value can only be null.
dns1	Yes	String	The value can only be null.
dns2	Yes	String	The value can only be null.
gateway	Yes	String	The value can only be null.
gateway_v6	Yes	String	The value can only be null.
netmask	Yes	String	The value can only be null.
netmask_v6	Yes	String	The value can only be null.
bridge	No	String	The value is fixed to be null and is in UUID format.

Example Request

Query the networks available to a tenant.

```
GET https://{endpoint}/v2.1/{project_id}/os-networks
```

Example Response

```
{
  "networks": [
    {
      "id": "04468f37-500a-4a80-88da-af823e7a1d6c",
      "cidr_v6": null,

```

```
"gateway": null,
"label": "network_demo1",
"broadcast": null,
"netmask": null,
"cidr": null,
"dns2": null,
"gateway_v6": null,
"netmask_v6": null,
"dns1": null
},
{
"id": "1fcff959-21d0-4ba8-976a-974cb564c977",
"cidr_v6": null,
"gateway": null,
"label": "network_demo2",
"broadcast": null,
"netmask": null,
"cidr": null,
"dns2": null,
"gateway_v6": null,
"netmask_v6": null,
"dns1": null
}
]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.4.2 Querying the Networks of a Specified ECS

Function

This API is used to query the networks of a specified ECS.

Constraints

None

URI

GET /v2.1/{project_id}/servers/{server_id}/ips

[Table 6-89](#) describes the parameters in the URI.

Table 6-89 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-90](#) describes the response parameters.

Table 6-90 Response parameters

Parameter	Mandatory	Type	Description
addresses	Yes	Object	Specifies the network address of the ECS. For details, see Table 6-91 .

Table 6-91 addresses parameter structure description

Parameter	Mandatory	Type	Description
Network address of the ECS	Yes	Array of objects	Specifies the network where the ECS accesses. For details about the network parameter, see Table 6-92 .

Table 6-92 ECS network parameter structure description

Attribute	Type	CRUD	Default Value	Constraint	Remarks
version	Integer	R	N/A	4 or 6	Specifies the IP address version. The value of this parameter can be 4 or 6 .
addr	String	R	N/A	IP address format	Specifies the IP address.

Example Request

Query the networks of a specified ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/ips
```

Example Response

```
{
  "addresses": {
    "Network address of the ECS": [
      {
        "version": 4,
        "addr": "10.176.42.16"
      },
      {
        "version": 6,
```

```
    "addr": "::babe:10.176.42.16"  
  }  
  ]  
}  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.4.3 Querying the Specified Network of an ECS

Function

This API is used to query the specified network of an ECS.

Constraints

None

URI

GET /v2.1/{project_id}/servers/{server_id}/ips/{networkName}

[Table 6-93](#) describes the parameters in the URI.

Table 6-93 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Table 6-94 Request parameters

Parameter	Mandatory	Type	Description
server_id	Yes	String	Specifies the ECS ID.
networkName	Yes	String	Specifies the ECS network name.

Request

None

Response

[Table 6-95](#) describes the response parameters.

Table 6-95 Response parameters

Parameter	Type	Description
Network address of the ECS	List(Dict)	Specifies the network where the ECS accesses. For details about the network, see Table 6-96 .

Table 6-96 ECS network parameter structure description

Attribute	Type	CRUD	Default Value	Constraint	Remarks
version	Integer	R	N/A	4 or 6	Specifies the IP address version. The value of this parameter can be 4 or 6 .
addr	String	R	N/A	IP address format	Specifies the IP address.

Example Request

Query the specified network of a specified ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/ips/{networkName}
```

Example Response

```
{
  "Network address of the ECS": [
    {
      "version": 4,
      "addr": "10.0.0.4"
    },
    {
      "version": 4,
      "addr": "192.150.73.132"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.5 Security Group Management

6.5.1 Adding a Security Group

Function

This API is used to add an ECS to a security group.

You are suggested to add an ECS to a maximum of five security groups.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-97](#) describes the parameters in the URI.

Table 6-97 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-98](#) describes the request parameters.

Table 6-98 Request parameter

Parameter	Mandatory	Type	Description
addSecurityGroup	Yes	Object	Specifies the security group where the ECS is added. For details, see Table 6-99 .

Table 6-99 addSecurityGroup parameter description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the UUID or name of the security group to which the ECS is added. The configuration takes effect for the NICs on the ECS.

Response

None

Example Request

Add a security group to a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "addSecurityGroup": {
    "name": "sg-test"
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.5.2 Removing a Security Group

Function

This API is used to remove a security group from an ECS.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-100](#) describes the parameters in the URI.

Table 6-100 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-101](#) describes the request parameters.

Table 6-101 Request parameter

Parameter	Mandatory	Type	Description
removeSecurityGroup	Yes	Object	Specifies the security group to be removed from an ECS. For details, see Table 6-102 .

Table 6-102 removeSecurityGroup parameter description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the UUID or name of the security group from which the ECS is removed. The configuration takes effect for the NICs on the ECS.

Response

None

Example Request

Remove a security group from an ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "removeSecurityGroup": {
    "name": "sg-test"
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.5.3 Querying Security Groups of a Specified ECS

Function

This API is used to query security groups of a specified ECS.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-security-groups

[Table 6-103](#) describes the parameters in the URI.

Table 6-103 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-104](#) describes the response parameters.

Table 6-104 Response parameters

Parameter	Mandatory	Type	Description
security_groups	Yes	Array of objects	Specifies security groups. For details, see Table 6-105 .

Table 6-105 security_group objects

Parameter	Mandatory	Type	Description
description	Yes	String	Specifies information about a security group. It must contain 0 to 255 characters.
id	Yes	String	Specifies the security group ID in UUID format.
name	Yes	String	Specifies the security group name. It must contain 0 to 255 characters.
rules	Yes	Array of objects	Specifies security group rules. For details, see Table 6-106 .
tenant_id	Yes	String	Specifies the tenant or project ID.

Table 6-106 security_group_rule objects

Parameter	Mandatory	Type	Description
parent_group_id	Yes	String	Specifies the associated security group ID in UUID format.
ip_protocol	Yes	String	Specifies the protocol type or the IP protocol number. The value can be icmp , tcp , udp , or the IP protocol number.
from_port	Yes	Integer	Specifies the start port number. The value ranges from 1 to 65,535 and cannot be greater than to_port . When ip_protocol is icmp , this parameter specifies a port type with a length from 0 to 255 characters.
to_port	Yes	Integer	Specifies the stop port number. The value ranges from 1 to 65,535 and cannot be less than from_port . When ip_protocol is icmp , it specifies the code. The value ranges from 0 to 255. If both from_port and to_port are -1 , any ICMP packet can be transmitted.
ip_range	Yes	Object	Specifies the peer IP segment in CIDR format. For details, see Table 6-107 . The value of ip_range or group must be empty.
group	Yes	Object	Specifies the name of the peer security group and the ID of the tenant in the peer security group. For details, see Table 6-108 . The value of ip_range or group must be empty.
id	Yes	String	Specifies the security group rule ID in UUID format.

Table 6-107 ip_range objects

Parameter	Mandatory	Type	Description
cidr	No	String	Specifies the peer IP segment in CIDR format.

Table 6-108 group objects

Parameter	Mandatory	Type	Description
tenant_id	No	String	Specifies the ID of the tenant of the peer security group.
name	No	String	Specifies the name of the peer security group.

Example Request

Query security groups of a specified ECS.

```
GET https://{endpoint}/v2.1/e73621affb8f44e1bc01898747ca09d4/servers/65fae4c2-3a09-46c6-af12-3b04f1fdb1e/os-security-groups
```

Example Response

```
{
  "security_groups": [
    {
      "rules": [
        {
          "from_port": null,
          "group": {
            "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
            "name": "default"
          },
          "ip_protocol": null,
          "to_port": null,
          "parent_group_id": "bc4ac1d1-dc77-4b7d-a97d-af86eb0dc450",
          "ip_range": {},
          "id": "bb3cc988-e06a-49f6-b668-600e8bf193ee"
        },
        {
          "from_port": null,
          "group": {
            "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
            "name": "default"
          },
          "ip_protocol": null,
          "to_port": null,
          "parent_group_id": "bc4ac1d1-dc77-4b7d-a97d-af86eb0dc450",
          "ip_range": {},
          "id": "f9371051-d7e1-4be4-8748-77b1e0913730"
        }
      ],
      "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
    }
  ]
}
```

```
"description": "default",
"id": "bc4ac1d1-dc77-4b7d-a97d-af86eb0dc450",
"name": "default"
},
{
  "rules": [
    {
      "from_port": 200,
      "group": {},
      "ip_protocol": "tcp",
      "to_port": 400,
      "parent_group_id": "b3e4b615-a40f-4e1c-92af-2e0d382141d5",
      "ip_range": {
        "cidr": "0.0.0.0/0"
      },
      "id": "3330120d-bbd1-4a73-bda9-0196a84d5670"
    },
    {
      "from_port": 201,
      "group": {},
      "ip_protocol": "tcp",
      "to_port": 400,
      "parent_group_id": "b3e4b615-a40f-4e1c-92af-2e0d382141d5",
      "ip_range": {
        "cidr": "0.0.0.0/0"
      },
      "id": "b550c9a6-970a-462d-984e-265e88020818"
    }
  ],
  "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
  "description": "desc-sg",
  "id": "b3e4b615-a40f-4e1c-92af-2e0d382141d5",
  "name": "test-sg"
}
]
```

Returned Values

See [Returned Values for General Requests](#).

6.6 Flavor Management

6.6.1 Querying ECS Flavors

Function

This API is used to query available ECS flavors. After receiving the request, Nova uses nova-api to view the flavors from the database.

URI

GET /v2.1/{project_id}/flavors?
minDisk={minDisk}&minRam={minRam}&sort_key={sort_key}&sort_dir={sort_dir}

[Table 6-109](#) describes the parameters in the URI.

Table 6-109 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Parameters in the following table can be used as URI parameters to filter query results. Usage: /v2/{project_id}/flavors?minDisk={minDisk}&minRam={minRam}

[Table 6-110](#) describes the query parameters.

Table 6-110 Query parameters

Parameter	Mandatory	Type	Description
minDisk	No	Integer	Specifies the minimum disk specification in the unit of GB. Only the ECSs with the disk specification greater than or equal to the minimum specification can be queried.
minRam	No	Integer	Specifies the minimum RAM in the unit of MB. Only the ECSs with the RAM specification greater than or equal to the minimum specification can be queried.
sort_key	No	String	Indicates a sorting field, the default value of which is flavorid . The value of this parameter can also be name , memory_mb , vcpus , root_gb , or flavorid .
sort_dir	No	String	Specifies the ascending (asc) or descending (desc) sorting. The default value is asc .

Request

None

Response

[Table 6-112](#) describes the response parameters.

Table 6-111 Response parameters

Parameter	Type	Description
flavors	Array of objects	Specifies ECS flavors. For details, see Table 6-112 .
flavors_links	Array of objects	Specifies data links for querying the next pages in pagination query. For details, see Table 6-113 .

Table 6-112 flavors field description

Parameter	Type	Description
id	String	Specifies the flavor ID.
links	Array of objects	Specifies the shortcut link of the ECS flavor. For details, see Table 6-113 .
name	String	Specifies the flavor name.

Table 6-113 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Specifies the shortcut link.

Example Request

Query available ECS flavors.

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors
```

Example Response

```
{
  "flavors": [
    {
      "id": "s3.xlarge.1",
      "links": [
        {
          "href": "https://compute.region.example.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/s3.xlarge.1",
          "rel": "self"
        },
        {
          "href": "https://compute.region.example.com/743b4c0428d94531b9f2add666642e6b/flavors/s3.xlarge.1",
          "rel": "bookmark"
        }
      ]
    },
    {
      "name": "s3.xlarge.1"
    }
  ]
}
```

```
    },
    {
      "id": "s3.xlarge.2",
      "links": [
        {
          "href": "https://compute.region.example.com/v2.1/743b4c0428d94531b9f2add666642e6b/
flavors/s3.xlarge.2",
          "rel": "self"
        },
        {
          "href": "https://compute.region.x.com/743b4c0428d94531b9f2add666642e6b/flavors/
s3.xlarge.2",
          "rel": "bookmark"
        }
      ],
      "name": "s3.xlarge.2"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.6.2 Querying Details About ECS Flavors

Function

This API is used to query details about ECS flavors.

URI

GET /v2.1/{project_id}/flavors/detail?
minDisk={minDisk}&minRam={minRam}&sort_key={sort_key}&sort_dir={sort_dir}

[Table 6-114](#) describes the parameters in the URI.

Table 6-114 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Table 6-115 Query parameters

Parameter	Mandatory	Type	Description
minDisk	No	String	Specifies the minimum disk specification in the unit of GB. Only the ECSs with the disk specification greater than or equal to the minimum specification can be queried.

Parameter	Mandatory	Type	Description
minRam	No	String	Specifies the minimum RAM in the unit of MB. Only the ECSs with the RAM specification greater than or equal to the minimum specification can be queried.
sort_key	No	String	Indicates a sorting field, the default value of which is flavorid . The value of this parameter can also be name , memory_mb , vcpus , root_gb , or flavorid .
sort_dir	No	String	Specifies the ascending (asc) or descending (desc) sorting. The default value is asc .

Request

None

Response

[Table 6-116](#) describes the response parameters.

Table 6-116 Response parameters

Parameter	Type	Description
flavors	Array of objects	Specifies ECS flavors. For details, see Table 6-117 .
flavors_links	Array of objects	Specifies data links for querying the next pages in pagination query. For details, see Table 6-118 .

Table 6-117 flavors field description

Parameter	Type	Description
id	String	Specifies the ID of the ECS flavor.
name	String	Specifies the name of the ECS flavor.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor.
ram	Integer	Specifies the memory size (MB) in the ECS flavor.
disk	Integer	Specifies the system disk size in the ECS flavor. This parameter has not been used. Its default value is 0 .

Parameter	Type	Description
swap	String	Specifies the swap partition size required by the ECS flavor. This parameter has not been used. Its default value is "".
OS-FLV-EXT-DATA:ephemeral	Integer	Specifies the temporary disk size. This is an extended attribute. This parameter has not been used. Its default value is 0 .
OS-FLV-DISABLED:disabled	Boolean	Specifies whether the ECS flavor has been disabled. This is an extended attribute. This parameter has not been used. Its default value is false .
rxtx_factor	Float	Specifies the ratio of the available network bandwidth to the network hardware bandwidth of the ECS. This parameter has not been used. Its default value is 1.0 .
os-flavor-access:is_public	Boolean	Specifies whether a flavor is available to all tenants. This is an extended attribute. <ul style="list-style-type: none">• true: indicates that a flavor is available to all tenants.• false: indicates that a flavor is available only to certain tenants. Default value: true
links	Array of objects	Specifies shortcut links for ECS flavors. For details, see Table 6-118 .

Table 6-118 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Provides the corresponding shortcut link.

Example Request

Query details about ECS flavors.

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/detail
```

Example Response

```
{
  "flavors": [
    {
      "name": "c3.2xlarge.2",
      "links": [
        {
          "href": "https://compute.region.example.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",
          "rel": "self"
        },
        {
          "href": "https://compute.region.example.com/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",
          "rel": "bookmark"
        }
      ],
      "ram": 16384,
      "OS-FLV-DISABLED:disabled": false,
      "vcpus": 8,
      "swap": "",
      "os-flavor-access:is_public": true,
      "rxtx_factor": 1,
      "OS-FLV-EXT-DATA:ephemeral": 0,
      "disk": 0,
      "id": "c3.2xlarge.2"
    },
    {
      "name": "c3.2xlarge.4",
      "links": [
        {
          "href": "https://compute.region.example.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.4",
          "rel": "self"
        },
        {
          "href": "https://compute.region.example.com/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.4",
          "rel": "bookmark"
        }
      ],
      "ram": 32768,
      "OS-FLV-DISABLED:disabled": false,
      "vcpus": 8,
      "swap": "",
      "os-flavor-access:is_public": true,
      "rxtx_factor": 1,
      "OS-FLV-EXT-DATA:ephemeral": 0,
      "disk": 0,
      "id": "c3.2xlarge.4"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.6.3 Querying Details About an ECS Flavor

Function

This API is used to query the details about an ECS flavor based on the flavor ID.

URI

GET /v2.1/{project_id}/flavors/{flavor_id}

[Table 6-119](#) describes the parameters in the URI.

Table 6-119 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
flavors_id	Yes	Specifies the flavor ID.

Request

None

Response

[Table 6-120](#) describes the response parameters.

Table 6-120 Response parameters

Parameter	Type	Description
flavor	Object	Specifies the ECS flavor. For details, see Table 6-121 .

Table 6-121 flavor field description

Parameter	Type	Description
id	String	Specifies the ID of the ECS flavor.
name	String	Specifies the name of the ECS flavor.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor.
ram	Integer	Specifies the memory size (MB) in the ECS flavor.
disk	Integer	Specifies the system disk size in the ECS flavor. This parameter has not been used. Its default value is 0 .
swap	String	Specifies the swap partition size required by the ECS flavor. This parameter has not been used. Its default value is "".

Parameter	Type	Description
OS-FLV-EXT-DATA:ephemeral	Integer	Specifies the temporary disk size. This is an extended attribute. This parameter has not been used. Its default value is 0 .
OS-FLV-DISABLED:disabled	Boolean	Specifies whether the ECS flavor has been disabled. This is an extended attribute. This parameter has not been used. Its default value is false .
rxtx_factor	Float	Specifies the ratio of the available network bandwidth to the network hardware bandwidth of the ECS. This parameter has not been used. Its default value is 1.0 .
os-flavor-access:is_public	Boolean	Specifies whether a flavor is available to all tenants. This is an extended attribute. <ul style="list-style-type: none">• true: indicates that a flavor is available to all tenants.• false: indicates that a flavor is available only to certain tenants. Default value: true
links	Array of objects	Specifies shortcut links for ECS flavors. For details, see Table 6-122 .

Table 6-122 links field description

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name.
href	String	Provides the corresponding shortcut link.

Example Request

Query details about an ECS flavor.

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2
```

Example Response

```
{
  "flavor": {
    "name": "c3.2xlarge.2",
    "links": [
      {
        "href": "https://compute.region.example.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",

```

```
    "rel": "self"
  },
  {
    "href": "https://compute.region.example.com/743b4c0428d94531b9f2add666642e6b/flavors/
c3.2xlarge.2",
    "rel": "bookmark"
  }
],
"ram": 16384,
"OS-FLV-DISABLED:disabled": false,
"vcpus": 8,
"swap": "",
"os-flavor-access:is_public": true,
"rxtx_factor": 1,
"OS-FLV-EXT-DATA:ephemeral": 0,
"disk": 0,
"id": "c3.2xlarge.2"
}
```

Returned Values

See [Returned Values for General Requests](#).

6.6.4 Querying the extra_specs Value for an ECS Flavor

Function

This API is used to query the **extra_specs** value for a specified ECS flavor.

URI

GET /v2.1/{project_id}/flavors/{flavor_id}/os-extra_specs

[Table 6-123](#) describes the parameters in the URI.

Table 6-123 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
flavors_id	Yes	Specifies the flavor ID.

Request

None

Response

[Table 6-124](#) describes the response parameters.

Table 6-124 Response parameters

Parameter	Type	Description
extra_specs	Map<String,String>	Specifies the key-value pair of an ECS flavor. For details about the returned fields, see the os_extra_specs field description in Table 5-41 .

Example Request

Query the extra_specs details of a specified ECS flavor.

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2/os-extra_specs
```

Example Response

```
{
  "extra_specs": {
    "ecs:performancetype": "computingv3",
    "resource_type": "IOoptimizedC3_2"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.7 NIC Management

6.7.1 Querying NICs of an ECS

Function

This API is used to query NICs attached to an ECS.

URI

```
GET /v2.1/{project_id}/servers/{server_id}/os-interface
```

[Table 6-125](#) describes the parameters in the URI.

Table 6-125 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-126](#) describes the response parameters.

Table 6-126 Response parameters

Parameter	Type	Description
interfaceAttachments	Array of objects	Specifies ECS NICs. For details, see Table 6-127 .

Table 6-127 interfaceAttachments field description

Parameter	Type	Description
port_state	String	Specifies the NIC port status.
fixed_ips	Array of objects	Specifies private IP addresses for NICs. For details, see Table 6-128 .
net_id	String	Specifies the network ID to which the NIC port belongs.
port_id	String	Specifies the ID of the NIC port.
mac_addr	String	Specifies the MAC address of the NIC.

Table 6-128 fixed_ips field description

Parameter	Type	Description
subnet_id	String	Specifies the subnet of the NIC private IP address.
ip_address	String	Specifies the NIC private IP address.

Example Request

Query NICs attached to an ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface
```

Example Response

```
{
  "interfaceAttachments": [
    {
      "port_state": "ACTIVE",
      "fixed_ips": [
        {
          "subnet_id": "f8a6e8f8-c2ec-497c-9f23-da9616de54ef",
          "ip_address": "192.168.1.3"
        }
      ],
      "net_id": "3cb9bc59-5699-4588-a4b1-b87f96708bc6",
      "port_id": "ce531f90-199f-48c0-816c-13e38010b442",
      "mac_addr": "fa:16:3e:4c:2c:30"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.7.2 Querying Details About a Specified NIC of an ECS

Function

This API is used to query details about an NIC based on the NIC ID.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-interface/{port_id}

[Table 6-129](#) describes the parameters in the URI.

Table 6-129 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
port_id	Yes	Specifies the port ID of the NIC.

Request

None

Response

[Table 6-130](#) describes the response parameters.

Table 6-130 Response parameters

Parameter	Type	Description
interfaceAttachment	Object	Specifies ECS NICs. For details, see Table 6-131 .

Table 6-131 interfaceAttachment field description

Parameter	Type	Description
port_state	String	Specifies the NIC port status.
fixed_ips	Array of objects	Specifies IP addresses for NICs. For details, see Table 6-132 .
net_id	String	Specifies the network ID to which the NIC port belongs.
port_id	String	Specifies the ID of the NIC port.
mac_addr	String	Specifies the MAC address of the NIC.

Table 6-132 fixed_ips field description

Parameter	Type	Description
subnet_id	String	Specifies the ID of the subnet used by the NIC.
ip_address	String	Specifies the NIC IP address.

Example Request

Query details about an NIC based on the specified NIC ID.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface/{port_id}
```

Example Response

```
{
  "interfaceAttachment":
  {
    "port_state": "ACTIVE",
    "fixed_ips": [
      {
        "subnet_id": "f8a6e8f8-c2ec-497c-9f23-da9616de54ef",
        "ip_address": "192.168.1.3"
      }
    ],
    "net_id": "3cb9bc59-5699-4588-a4b1-b87f96708bc6",
    "port_id": "ce531f90-199f-48c0-816c-13e38010b442",
    "mac_addr": "fa:16:3e:4c:2c:30"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.7.3 Adding a NIC to an ECS

Function

This API is used to add a NIC to an ECS.

URI

POST /v2.1/{project_id}/servers/{server_id}/os-interface

[Table 6-133](#) describes the parameters in the URI.

Table 6-133 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-134](#) describes the request parameters.

Table 6-134 Request parameters

Parameter	Mandatory	Type	Description
interfaceAttachment	Yes	Object	Specifies the NICs to be added. For details, see Table 6-135 .

Table 6-135 interfaceAttachment field description

Parameter	Mandatory	Type	Description
port_id	No	String	Specifies the port ID. The port_id and net_id cannot be specified at the same time.
net_id	No	String	Specifies the network ID. The port_id and net_id cannot be specified at the same time.

Parameter	Mandatory	Type	Description
fixed_ips	No	Array of objects	Specifies a private IP address. This parameter cannot be specified when port_id is used. This parameter must be used with net_id . Only the first element in the list is valid. If two or more elements are used, an error will be reported. For details, see Table 6-136 .

Table 6-136 fixed_ips field description

Parameter	Mandatory	Type	Description
ip_address	No	String	Specifies the IP address.

Response

[Table 6-137](#) describes the response parameters.

Table 6-137 Response parameters

Parameter	Type	Description
interfaceAttachment	Object	Specifies ECS NICs. For details, see Table 6-138 .

Table 6-138 interfaceAttachment field description

Parameter	Type	Description
port_state	String	Specifies the port state.
fixed_ips	Array of objects	Specifies IP addresses for NICs. For details, see Table 6-139 .
port_id	String	Specifies the port ID.
net_id	String	Specifies the network ID.
mac_addr	String	Specifies the MAC address.

Table 6-139 fixed_ips field description

Parameter	Type	Description
subnet_id	String	Specifies the ID of the subnet used by the NIC.
ip_address	String	Specifies the NIC IP address.

Example Request

- Add a NIC whose network ID is **3cb9bc59-5699-4588-a4b1-b87f96708bc6** to a specified ECS.

POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface

```
{
  "interfaceAttachment": {
    "fixed_ips": [
      {
        "ip_address": "192.168.1.3"
      }
    ],
    "net_id": "3cb9bc59-5699-4588-a4b1-b87f96708bc6"
  }
}
```

- Add a NIC whose port ID is **ce531f90-199f-48c0-816c-13e38010b442** to a specified ECS.

POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface

```
{
  "interfaceAttachment": {
    "fixed_ips": [
      {
        "ip_address": "192.168.1.3"
      }
    ],
    "port_id": "ce531f90-199f-48c0-816c-13e38010b442"
  }
}
```

Example Response

```
{
  "interfaceAttachment": {
    "port_state": "DOWN",
    "fixed_ips": [
      {
        "subnet_id": "d9cfef77-0151-4c2a-9ed5-d951ada8adf3",
        "ip_address": "10.0.1.11"
      }
    ],
    "port_id": "ce531f90-199f-48c0-816c-13e38010b442",
    "net_id": "0dc714fa-9022-4a03-bb22-9821a396bb9d",
    "mac_addr": "fa:16:3e:63:75:b2"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.7.4 Deleting a NIC from an ECS

Function

This API is used to delete a NIC from an ECS based on the port ID.

Constraints

The primary NIC of an ECS has routing rules configured and cannot be deleted.

When an ECS NIC is detached, the NIC that is attached to the ECS and specified by **port_id** through the OpenStack Nova API will be retained, and the NIC specified by **net_id** will be deleted.

URI

DELETE /v2.1/{project_id}/servers/{server_id}/os-interface/{port_id}

[Table 6-140](#) describes the parameters in the URI.

Table 6-140 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
port_id	Yes	Specifies the port ID of the NIC. NOTE When the ID is the same as the ECS primary NIC ID, the system will return error code 403.

Request

None

Response

None

Example Request

Delete a NIC from an ECS based on the specified port ID.

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface/{port_id}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.8 Disk Management

6.8.1 Querying Disks Attached to an ECS

Function

This API is used to query the disks that have been and are being attached to an ECS.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments

[Table 6-141](#) describes the parameters in the URI.

Table 6-141 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

Response parameters

[Table 6-142](#) describes the response parameters.

Table 6-142 Response parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies the disks attached to an ECS. For details, see Table 6-143 .

Table 6-143 volumeAttachments field description

Parameter	Type	Description
device	String	Specifies the attached directory.
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ECS ID.
volumeId	String	Specifies the ID of the attached disk.

Example Request

Query details about disks attached to an ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-volume_attachments
```

Example Response

```
{
  "volumeAttachments": [
    {
      "device": "/dev/sdd",
      "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
      "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
      "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
    },
    {
      "device": "/dev/sdc",
      "id": "a26887c6-c47b-4654-abb5-dfadf7d3f804",
      "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
      "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f804"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.8.2 Querying a Disk Attached to an ECS

Function

This API is used to query a disk attached to an ECS based on the disk ID.

URI

```
GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}
```

[Table 6-144](#) describes the parameters in the URI.

Table 6-144 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
volume_id	Yes	Specifies the disk ID.

Request

None

Response

[Table 6-145](#) describes the response parameters.

Table 6-145 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disks attached to an ECS. For details, see Table 6-146 .

Table 6-146 volumeAttachment field description

Parameter	Type	Description
device	String	Specifies the attached directory.
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ECS ID.
volumeId	String	Specifies the ID of the attached disk.

Example Request

Query details about a disk attached to an ECS based on the specified disk ID.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}
```

Example Response

```
{
  "volumeAttachment":
  {
    "device": "/dev/sdd",
    "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
```



```
"serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",  
"volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.8.3 Attaching an ECS Data Disk

Function

This API is used to attach a disk to an ECS.

Constraints

1. If you attach a bootable disk to an ECS, you must specify the disk drive letter.
2. A disk created using a backup cannot be attached to an ECS as the system disk.
3. An ECS in the **SUSPENDED** or **PAUSED** state, which is specified using the **OS-EXT-STS:vm_state** parameter of the ECS, cannot have a disk attached.
4. The EVS must be in the **available** status.
5. The EVS disk and the target ECS must be located in the same AZ.
6. VBD EVS disks cannot be attached to BMSs.

URI

POST /v2.1/{project_id}/servers/{server_id}/os-volume_attachments

[Table 6-147](#) describes the parameters in the URI.

Table 6-147 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-148](#) describes the request parameters.

Table 6-148 Request parameters

Parameter	Mandatory	Type	Description
volumeAttachment	Yes	Object	Specifies the volumes to be attached. For details, see Table 6-149 .

Table 6-149 volumeAttachment field description

Parameter	Mandatory	Type	Description
volumeId	Yes	String	Specifies the ID of the disk to be attached. The value is in UUID format.
device	No	String	<p>Specifies the device name, such as <code>/dev/sda</code> or <code>/dev/sdb</code>.</p> <p>The new disk device name cannot be the same as an existing one.</p> <p>The device name must be specified based on the sequence of existing device names. Otherwise, the system automatically generates one.</p> <p>NOTE VBD disk device names can only be <code>/dev/vdb</code> through <code>/dev/vdx</code>. You are advised to attach the VBD disks in alphabetical order. Otherwise, the disk drive letters may be incorrect on the ECS.</p>

Response

[Table 6-150](#) describes the response parameters.

Table 6-150 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disks attached to an ECS. For details, see Table 6-151 .

Table 6-151 volumeAttachment field description

Parameter	Type	Description
device	String	Specifies the device name.
serverId	String	Specifies the ID of the target ECS in UUID format.
id	String	Specifies the disk ID in UUID format.

Parameter	Type	Description
volumeld	String	Specifies the attaching ID, which is the same as the UUID.

Example Request

Attach the disk whose ID is **54667652-3029-4af8-9222-2d53066fd61c** to **/dev/sdb** of a specified ECS.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-volume_attachments
```

```
{
  "volumeAttachment": {
    "volumeld": "54667652-3029-4af8-9222-2d53066fd61c",
    "device": "/dev/sdb"
  }
}
```

Example Response

```
{
  "volumeAttachment": {
    "device": "/dev/vdb",
    "serverId": "ab258e25-e351-47c7-b6e3-0749c5d9ed6a",
    "id": "54667652-3029-4af8-9222-2d53066fd61c",
    "volumeld": "54667652-3029-4af8-9222-2d53066fd61c"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.8.4 Detaching a Disk from an ECS

Function

This API is used to detach a disk from an ECS.

Constraints

The system disk, the device name of which is **/dev/sda**, and user disks can be detached from an ECS only when the ECS is stopped. There are no requirements on the OSs and UVP VMTools.

When an ECS is in the **active** status, pay attention to the following constraints:

1. Only data disks, the device name of which is not **/dev/sda**, can be detached from an ECS.
2. Make sure that UVP VMTools have been installed and enabled on the ECS. Otherwise, the uninstallation will fail.
3. For a Linux ECS, you need to log in to the ECS and run the **umount** command to disassociate the target disk from the file system. In addition, you need to ensure that no data is being written into or being read from the disk. Otherwise, the detachment will fail.

4. For a Windows ECS, you need to ensure that no data is being written into or being read from the disk when a disk is to be detached from the running ECS. Otherwise, data will be lost.
5. OSs supporting EVS disk detachment from a running ECS include two parts:
 - For the first part, see [External Image File Formats and Supported OSs](#).
 - [Table 6-152](#) lists the second part of supported OSs.

Table 6-152 OSs supporting EVS disk detachment from a running ECS

OS	Version
CentOS	7.3 64bit
	7.2 64bit
	6.8 64bit
	6.7 64bit
Debian	8.6.0 64bit
	8.5.0 64bit
Fedora	25 64bit
	24 64bit
SUSE	SUSE Linux Enterprise Server 12 SP2 64bit
	SUSE Linux Enterprise Server 12 SP1 64bit
	SUSE Linux Enterprise Server 11 SP4 64bit
	SUSE Linux Enterprise Server 12 64bit
OpenSUSE	42.2 64bit
	42.1 64bit
Oracle Linux Server release	7.3 64bit
	7.2 64bit
	6.8 64bit
	6.7 64bit
Ubuntu Server	16.04 64bit
	14.04 64bit
	14.04.4 64bit
Windows (SCSI EVS disks cannot be detached from a running ECS.)	Windows Server 2008 R2 Enterprise 64bit
	Windows Server 2012 R2 Standard 64bit
	Windows Server 2016 R2 Standard 64bit

OS	Version
Red Hat Linux Enterprise	7.3 64bit
	6.8 64bit

- The forcible online disk detach function supports only VBD disks used by KVM ECSs.
For other types of disks used by Xen ECSs, BMSs, and KVM ECSs, this API supports only online disk detachment.
- Disks which are forcibly detached online will use the disk drives and PCI addresses, so the disk drives and PCI addresses will not be assigned again.
- After a disk is forcibly detached, it still occupies the disk quota of the ECS.
- The system disk cannot be detached forcibly online.
- When a file system is attached to a disk and the disk is detached forcibly online, users need to manually detach all file systems attached to the disk.
- If logical partitions are created on the disk which is detached forcibly online, the logical partitions will become invalid.
- After a disk is forcibly detached, you need to restart the ECS to clear the residual.

URI

DELETE /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}{?delete_flag}

Table 6-153 describes the parameters in the URI.

Table 6-153 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
volume_id	Yes	Specifies the volume ID.

Usage: DELETE /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}?delete_flag=1

Table 6-154 describes the query parameters.

Table 6-154 Query parameters

Parameter	Mandatory	Type	Description
delete_flag	No	Integer	Specifies whether to support forcible online disk detachment. The default value is 0 . 1 indicates that the disk can be forcibly detached online.

Request

None

Response

None

Example Request

Detach the disk whose ID is **54667652-3029-4af8-9222-2d53066fd61c** from a specified ECS.

```
DELETE https://{endpoint}/v2.1/6f9e9263116a4b68818cf1edce16bc4f/servers/ab258e25-e351-47c7-b6e3-0749c5d9ed6a/os-volume_attachments/54667652-3029-4af8-9222-2d53066fd61c
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.9 Metadata Management

6.9.1 Updating ECS Metadata

Function

This API is used to update ECS metadata.

- If the metadata does not contain the target field, the field is automatically added.
- If the metadata contains the target field, the field value is automatically updated.
- If the field in the metadata is not requested, the field value remains unchanged.

Constraints

An ECS must be in active, stopped, paused, or suspended state, which is specified by **OS-EXT-STS:vm_state**.

URI

POST /v2.1/{project_id}/servers/{server_id}/metadata

[Table 6-155](#) describes the parameters in the URI.

Table 6-155 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-156](#) describes the request parameters.

Table 6-156 Request parameters

Parameter	Mandatory	Type	Description
metadata	Yes	Object	Specifies the user-defined metadata key-value pair. For a metadata key: It contains a maximum of 255 Unicode characters and cannot be left blank. A key can contain uppercase letters (A-Z), lowercase letters (a-z), digits (0-9), hyphens (-), underscores (_), colons (:), and periods (.). For a metadata value: A value contains a maximum of 255 Unicode characters.

Response

[Table 6-157](#) describes the response parameters.

Table 6-157 Response parameters

Parameter	Type	Description
metadata	Object	Specifies the user-defined metadata key-value pair.

Example Request

Update the metadata of a specified ECS to the user-defined metadata key-value pair.

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata
```

```
{
  "metadata": {
    "key": "value"
  }
}
```

Example Response

```
{
  "metadata": {
    "key": "value"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.9.2 Configuring ECS Metadata

Function

This API is used to configure ECS metadata.

When you call this API, all the metadata of this ECS will be deleted, and the ECS uses the value configured in the request.

Constraints

An ECS must be in active, stopped, paused, or suspended state, which is specified by **OS-EXT-STS:vm_state**.

URI

```
PUT /v2.1/{project_id}/servers/{server_id}/metadata
```

[Table 6-158](#) describes the parameters in the URI.

Table 6-158 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-159](#) describes the request parameters.

Table 6-159 Request parameters

Parameter	Type	Mandatory	Description
metadata	Object	Yes	Specifies the user-defined metadata key-value pair. For a metadata key: A key contains a maximum of 255 Unicode characters and cannot be empty. A key can contain uppercase letters (A-Z), lowercase letters (a-z), digits (0-9), hyphens (-), underscores (_), colons (:), and periods (.). For a metadata value: A value contains a maximum of 255 Unicode characters.

Response

[Table 6-160](#) describes the response parameters.

Table 6-160 Response parameters

Parameter	Type	Description
metadata	Object	Specifies the user-defined metadata key-value pair.

Example Request

Configure the metadata of a specified ECS.

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata
{
  "metadata": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Example Response

```
{
  "metadata": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.9.3 Deleting Specified ECS Metadata

Function

This API is used to delete specified ECS metadata.

Constraints

An ECS must be in active, stopped, paused, or suspended state, which is specified by **OS-EXT-STS:vm_state**.

URI

DELETE /v2.1/{project_id}/servers/{server_id}/metadata/{key}

[Table 6-161](#) describes the parameters in the URI.

Table 6-161 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
key	Yes	Specifies the ECS metadata key.

Request

None

Response

None

Example Request

Delete the metadata from a specified ECS.

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata/{key}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.9.4 Querying ECS Metadata

Function

This API is used to query ECS metadata.

URI

GET /v2.1/{project_id}/servers/{server_id}/metadata

[Table 6-162](#) describes the parameters in the URI.

Table 6-162 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

NOTE

Pagination query is not supported.

Request

None

Response

[Table 6-163](#) describes the response parameters.

Table 6-163 Response parameters

Parameter	Mandatory	Type	Description
metadata	Yes	Object	Specifies the user-defined metadata key-value pair.

Example Request

Query metadata details of a specified ECS.

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/998af54b-5762-4041-abc1-f98a2c27b3a2/metadata
```

Example Response

```
{
  "metadata": {
    "wj": "True"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.9.5 Obtaining ECS Metadata with a Specified Key

Function

This API is used to obtain ECS metadata with a specified key.

URI

```
GET /v2.1/{project_id}/servers/{server_id}/metadata/{key}
```

[Table 6-164](#) describes the parameters in the URI.

Table 6-164 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
key	Yes	Specifies the ECS metadata key.

Request

None

Response

[Table 6-165](#) describes the response parameters.

Table 6-165 Response parameters

Parameter	Type	Description
meta	Object	Specifies the user-defined metadata key-value pair.

Example Request

Obtain the metadata with a specified key of an ECS.

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/998af54b-5762-4041-abc1-f98a2c27b3a2/metadata/key1
```

Example Response

```
{
  "meta": {
    "key1": "value1"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.9.6 Modifying ECS Metadata with a Specified Key

Function

This API is used to modify the ECS metadata with a specified key.

- If the metadata does not contain the target field, the field is automatically added.
- If the metadata contains the target field, the field value is automatically updated.

Constraints

An ECS must be in active, stopped, paused, or suspended state, which is specified by **OS-EXT-STS:vm_state**.

URI

```
PUT /v2.1/{project_id}/servers/{server_id}/metadata/{key}
```

[Table 6-166](#) describes the parameters in the URI.

Table 6-166 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
key	Yes	Specifies the ECS metadata key.

Request

[Table 6-167](#) describes the request parameters.

Table 6-167 Request parameters

Parameter	Mandatory	Type	Description
meta	Yes	Object	Specifies the user-defined metadata key pair. For a metadata key: It contains a maximum of 255 Unicode characters and cannot be left blank. A key can contain uppercase letters (A-Z), lowercase letters (a-z), digits (0-9), hyphens (-), underscores (_), colons (:), and periods (.). For a metadata value: It contains a maximum of 255 Unicode characters.

Response

[Table 6-168](#) describes the response parameters.

Table 6-168 Response parameters

Parameter	Type	Description
meta	Object	Specifies the user-defined metadata key-value pair.

Example Request

Set the metadata with a specified key of an ECS.

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata/{key}

{
  "meta":{
    "key":"value"
  }
}
```

Example Response

```
{
  "meta":{
    "key":"value"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.10 Tenant Quota Management

6.10.1 Querying Tenant Quota Limits

Function

This API is used to query tenant quota limits.

Tenants are only allowed to query their own quota limits.

URI

GET /v2.1/{project_id}/limits?project_id={project_id}

[Table 6-169](#) describes the parameters in the URI.

Table 6-169 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

[Table 6-170](#) describes the response parameters.

Table 6-170 Response parameters

Parameter	Type	Description
limits	Object	Specifies tenant limits. For details, see Table 6-171 .

Table 6-171 limits parameter information

Parameter	Type	Description
rate	List	The value is empty.
absolute	Object	Specifies tenant quota limits. For details, see Table 6-172 .

Table 6-172 absolute parameter information

Parameter	Type	Description
maxServerMeta	String	Specifies the limit of ECS metadata quantity. If the value is -1 , there is no quantity limit.
maxPersonality	String	Specifies the quantity limit of injected files. If the value is -1 , there is no quantity limit.
totalServerGroupsUsed	String	Specifies the number of used ECS groups.
maxImageMeta	String	Specifies the limit of the image metadata quantity. If the value is -1 , there is no quantity limit.
maxPersonalitySize	String	Specifies the size limit of injected files. If the value is -1 , there is no size limit.

Parameter	Type	Description
maxTotalRAMSize	String	Specifies the total memory size limit. If the value is -1 , there is no size limit.
maxTotalKeypairs	String	Specifies the limit of key pair quantity. If the value is -1 , there is no quantity limit.
maxSecurityGroupRules	String	Specifies the maximum number of security group rules. If the value is -1 , there is no quantity limit. This parameter is not supported in microversion 2.35 and later.
maxServerGroups	String	Specifies the maximum number of ECS groups. If the value is -1 , there is no quantity limit.
totalCoresUsed	String	Specifies the number of used cores.
totalRAMUsed	String	Specifies the size of used memory.
maxSecurityGroups	String	Specifies the maximum number of security groups. If the value is -1 , there is no quantity limit.
totalFloatingIpsUsed	String	Specifies the number of used floating IP addresses.
totalInstancesUsed	String	Specifies the number of used ECSs.
totalSecurityGroupsUsed	String	Specifies the number of used security groups.
maxTotalFloatingIps	String	Specifies the maximum number of floating IP addresses. If the value is -1 , there is no quantity limit.
maxTotalInstances	String	Specifies the maximum number of ECSs. If the value is -1 , there is no quantity limit.

Parameter	Type	Description
maxTotalCores	String	Specifies the maximum number of cores. If the value is -1 , there is no quantity limit.
maxServerGroupMembers	String	Specifies the maximum number of members in an ECS group. If the value is -1 , there is no quantity limit.

Example Request

Query tenant quota limits.

```
GET https://{endpoint}/v2.1/d9ebe43510414ef590a4aa158605329e/limits
```

Example Response

```
{
  "limits": {
    "rate": [],
    "absolute": {
      "maxServerMeta": 128,
      "maxPersonality": 5,
      "totalServerGroupsUsed": 0,
      "maxImageMeta": 128,
      "maxPersonalitySize": 10240,
      "maxTotalRAMSize": 25165824,
      "maxTotalKeypairs": -1,
      "maxSecurityGroupRules": 20,
      "maxServerGroups": -1,
      "totalCoresUsed": 0,
      "totalRAMUsed": 0,
      "maxSecurityGroups": 10,
      "totalFloatingIpsUsed": 0,
      "totalInstancesUsed": 0,
      "totalSecurityGroupsUsed": 0,
      "maxTotalFloatingIps": 10,
      "maxTotalInstances": 2048,
      "maxTotalCores": 20480,
      "maxServerGroupMembers": -1
    }
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.10.2 Querying Tenant Quotas

Function

This API is used to query quotas, including ECSs, vCPUs, and memory.

This API provides the **user_id** parameter for obtaining the quota configuration of a specified user.

URI

GET /v2.1/{project_id}/os-quota-sets/{project_id}?user_id={user_id}

[Table 6-173](#) describes the parameters in the URI.

Table 6-173 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. If the specified project does not exist, the default quota in the system is returned.
user_id	No	Specifies the user ID. If the specified user does not exist, the default quota in the system is returned.

Request

None

Response

[Table 6-174](#) describes the response parameters.

Table 6-174 Response parameters

Parameter	Type	Description
quota_set	Object	Specifies the quota_set object. For details, see Table 6-175 .

Table 6-175 **quota_set** parameter description

Parameter	Type	Description
cores	Integer	Specifies the quantity quota of vCPUs.
fixed_ips	Integer	Specifies the quantity quota of fixed IP addresses. This parameter is not supported.
floating_ips	Integer	Specifies the quantity quota of floating IP addresses. This parameter is not supported.

Parameter	Type	Description
id	String	Specifies the project UUID.
injected_file_content_bytes	Integer	Specifies the size quota (bytes) of the files to be injected.
injected_file_path_bytes	Integer	Specifies the size quota (bytes) of the path for the files to be injected. This parameter is not supported currently.
injected_files	Integer	Specifies the quantity quota of the files to be injected.
instances	Integer	Specifies the quantity quota of ECSs.
key_pairs	Integer	Specifies the quantity quota of key pairs. This parameter is not supported.
metadata_items	Integer	Specifies the metadata quantity quota.
ram	Integer	Specifies the memory quota (MB).
security_group_rules	Integer	Specifies the quota of security group rules. This parameter is not supported.
security_groups	Integer	Specifies the quantity quota of security groups. This parameter is not supported.
server_groups	Integer	Specifies the quantity quota of ECS groups.
server_group_members	Integer	Specifies the ECS quota in an ECS group.

Example Request

Query quotas of resources such as ECSs, vCPUs, and memory.

```
GET https://{endpoint}/v2.1/d9ebe43510414ef590a4aa158605329e/os-quota-sets/d9ebe43510414ef590a4aa158605329e
```

Example Response

```
{
  "quota_set": {
    "cores": 20,
    "fixed_ips": 40,
    "floating_ips": 10,
    "id": "d9ebe43510414ef590a4aa158605329e",
    "injected_file_content_bytes": 10240,
    "injected_file_path_bytes": 255,
    "injected_files": 5,
```

```
"instances": 20,  
"key_pairs": 100,  
"metadata_items": 128,  
"ram": 51200,  
"security_group_rules": 20,  
"security_groups": 50,  
"server_group_members": 10,  
"server_groups": 10  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.10.3 Querying Default Quotas

Function

This API is used to query default quotas.

URI

GET /v2.1/{project_id}/os-quota-sets/{project_id}/defaults

[Table 6-176](#) describes the parameters in the URI.

Table 6-176 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

[Table 6-177](#) describes the response parameters.

Table 6-177 Response parameters

Parameter	Type	Description
quota_set	Object	Specifies the quota_set object. For details, see Table 6-178 .

Table 6-178 quota_set parameter description

Parameter	Type	Description
cores	Integer	Specifies the quantity quota of vCPUs.
fixed_ips	Integer	Specifies the quantity quota of fixed IP addresses. This parameter is not supported.
floating_ips	Integer	Specifies the quantity quota of floating IP addresses. This parameter is not supported.
id	String	Specifies the project UUID.
injected_file_content_bytes	Integer	Specifies the size quota (bytes) of the files to be injected.
injected_file_path_bytes	Integer	Specifies the size quota (bytes) of the path for the files to be injected.
injected_files	Integer	Specifies the quantity quota of the files to be injected.
instances	Integer	Specifies the quantity quota of ECSs.
key_pairs	Integer	Specifies the quota of key pairs. This parameter is not supported.
metadata_items	Integer	Specifies the metadata quantity quota.
ram	Integer	Specifies the memory quota (MB).
security_group_rules	Integer	Specifies the quota of security group rules. This parameter is not supported.
security_groups	Integer	Specifies the quota of security groups. This parameter is not supported.
server_groups	Integer	Specifies the quantity quota of ECS groups.
server_group_members	Integer	Specifies the ECS quota in an ECS group.

Example Request

Query default quotas of resources.

```
GET https://{endpoint}/v2.1/d9ebe43510414ef590a4aa158605329e/os-quota-sets/  
d9ebe43510414ef590a4aa158605329e/defaults
```

Example Response

```
{  
  "quota_set":{  
    "injected_file_content_bytes":10240,  
    "metadata_items":128,  
    "server_group_members":10,  
    "server_groups":10,  
    "ram":51200,  
    "floating_ips":10,  
    "key_pairs":100,  
    "injected_file_path_bytes":255,  
    "instances":10,  
    "security_group_rules":20,  
    "injected_files":5,  
    "cores":20,  
    "fixed_ips":-1,  
    "id":"474eff20eee84b2e87b5717cc7f34dd8",  
    "security_groups":10  
  }  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.11 Key and Password Management

6.11.1 Querying SSH Key Pairs

Function

This API is used to query SSH key pairs.

URI

```
GET /v2.1/{project_id}/os-keypairs
```

[Table 6-179](#) describes the parameters in the URI.

Table 6-179 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Returned Values

See [Returned Values for General Requests](#).

6.11.2 Querying a Specified SSH Key Pair

Function

This API is used to query a specified SSH key pair based on the SSH key pair name.

URI

GET /v2.1/{project_id}/os-keypairs/{keypair_name}

[Table 6-183](#) describes the parameters in the URI.

Table 6-183 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
keypair_name	Yes	Specifies the key pair name.

Request

None

Response

[Table 6-184](#) describes the response parameters.

Table 6-184 Response parameters

Parameter	Type	Description
keypair	Object	Specifies the SSH key pair. For details, see Table 6-185 .

Table 6-185 keypair field description

Parameter	Type	Description
public_key	String	Specifies information about the public key.
name	String	Specifies the key pair name.

Parameter	Type	Description
fingerprint	String	Specifies fingerprint information about the key pair.
created_at	String	Specifies the time when the key pair was created.
deleted	Boolean	Specifies whether a key pair has been deleted. <ul style="list-style-type: none"> true: indicates that the key has been deleted. false: indicates that the key is not deleted.
deleted_at	String	Specifies the time when the key pair was deleted.
id	Integer	Specifies the key pair ID.
updated_at	String	Specifies the time when the key pair was updated.
user_id	String	Specifies information about the user to which the key pair belongs.
type	String	Specifies the key type, which is ssh by default. This parameter is supported in microversion 2.2 and later.

Example Request

Query details of a specified SSH key pair.

```
GET https://{endpoint}/v2.1/{project_id}/os-keypairs/{keypair_name}
```

Example Response

```
{
  "keypair": {
    "created_at": "2014-05-07T12:06:13.681238",
    "deleted": false,
    "deleted_at": null,
    "fingerprint": "9d:00:f4:d7:26:6e:52:**:**:**:**:**:**:**",
    "id": 1,
    "name": "keypair-3582d8b7-e588-4aad-b7f7-f4e76f0e4314",
    "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDYJrTVpcMwFqQy/
oMvtUSRofZdSRHEwrsX8AYkRvn2ZnCXm+b6+GZ2NQuuWj+oczlnwiGFQDsL/yeE+/
kurqcPJFKKp60mToXIMyzioFxW88fjtw*****
*****
Generated-by-Nova\n",
    "updated_at": null,
    "user_id": "fake"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.11.3 Creating and Importing an SSH Key Pair

Function

This API is used to create an SSH key pair or import a public key to generate a key pair.

After a private SSH key is created, download the private key to a local directory. Then, you can use this private key to log in to the ECS. To ensure ECS security, the private key can be downloaded only once. Keep it secure.

Only the user that created the key pair can view it. If the key pair is created by an IAM user, the IAM account of the user and the other users of the same account cannot view the key pair.

URI

POST /v2.1/{project_id}/os-keypairs

[Table 6-186](#) describes the parameters in the URI.

Table 6-186 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 6-187](#) describes the request parameters.

 **NOTE**

When creating an SSH key, you only need to configure **name**. When importing an SSH key, you must configure **public_key**.

Table 6-187 Request parameters

Parameter	Mandatory	Type	Description
keypair	Yes	Object	Specifies the created or imported SSH key pair. For details, see Table 6-188 .

Table 6-188 keypair field description

Parameter	Mandatory	Type	Description
public_key	No	String	Specifies the imported public key. It is recommended that the length of the imported public key be less than or equal to 1024 bytes. NOTE If the length of the public key to be imported exceeds 1024 bytes, importing the public key will fail.
type	No	String	Specifies the key type. The value is ssh or x509 . This parameter is supported in microversion 2.2 and later.
name	Yes	String	Specifies the key pair name. The new key pair name cannot be the same as an existing one.
user_id	No	String	Specifies the user ID of the key. This parameter is supported in microversion 2.10 and later.

Response

[Table 6-189](#) describes the response parameters.

Table 6-189 Response parameters

Parameter	Type	Description
keypair	Object	Specifies the SSH key pair. For details, see Table 6-190 .

Table 6-190 keypair field description

Parameter	Type	Description
fingerprint	String	Specifies fingerprint information about the key pair.
name	String	Specifies the key pair name.
public_key	String	Specifies information about the public key.

Request

None

Response

None

Example Request

Delete the SSH key pair named **KeyPair-123**.

```
DELETE https://{endpoint}/v2.1/{project_id}/os-keypairs/KeyPair-123
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.11.5 Obtaining the Password of an ECS

Function

This API is used to obtain the random password generated during initial Windows ECS installation for user **Administrator** or the configured **Cloudbase-init** user when you use an image that supports Cloudbase-Init to create a Windows ECS.

After starting an ECS, wait for 5 to 10 minutes and ensure that the password is injected. Then, you can use this API to query the password.

Linux ECSs do not use this API to obtain a password.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-server-password

[Table 6-192](#) describes the parameters in the URI.

Table 6-192 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-193](#) describes the response parameters.

Table 6-193 Response parameters

Parameter	Type	Description
password	String	Specifies the password in ciphertext.

Example Request

Obtain a random password of the administrator account (administrator or the account configured in Cloudbase-Init) generated by the system during the initial installation of a specified Windows ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-server-password
```

Example Response

```
{
  "password": "UHC9+YW1xDC1Yu8Mg*****"
}
```

Returned Values

See [Returned Values for General Requests](#).

6.11.6 Deleting the Password of an ECS

Function

This API is used to delete the random password records generated during initial installation of a Windows ECS. After the password is deleted, you can still use your password to log in to your ECS. However, you cannot use the **Get Password** function to recover the ECS initial password.

Linux ECSs do not use this API to delete a password.

URI

```
DELETE /v2.1/{project_id}/servers/{server_id}/os-server-password
```

[Table 6-194](#) describes the parameters in the URI.

Table 6-194 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

Delete the password records generated during initial installation of a Windows ECS.

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-server-password
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.12 ECS Group Management

6.12.1 Creating an ECS Group

Function

This API is used to create an ECS group.

Constraints

Only anti-affinity groups are supported.

URI

POST /v2.1/{project_id}/os-server-groups

[Table 6-195](#) describes the parameters in the URI.

Table 6-195 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 6-196](#) describes the request parameters.

Table 6-196 Request parameters

Parameter	Mandatory	Type	Description
server_group	Yes	Object	Specifies the ECS group information. For details, see Table 6-197 .

Table 6-197 server_group field description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the ECS group name. The value contains 1 to 255 characters.
policies	Yes	Array of strings	Specifies the policies associated with the ECS group. Options: <ul style="list-style-type: none">anti-affinity: ECSs in this group must be deployed on different hosts. NOTE

Response

[Table 6-198](#) describes the response parameters.

Table 6-198 Response parameters

Parameter	Type	Description
server_group	Object	Specifies the ECS group information. For details, see Table 6-199 .

Table 6-199 server_group field description

Parameter	Type	Description
id	String	Specifies the ECS group UUID.
name	String	Specifies the ECS group name.
policies	Array of strings	Specifies the policies associated with the ECS group. Options: anti-affinity : ECSs in this group must be deployed on different hosts.
members	Array of strings	Specifies the ECSs contained in an ECS group.
metadata	Object	Specifies the ECS group metadata.
project_id	String	Specifies the tenant ID in UUID format for the ECS group. This parameter is supported in microversion 2.13 and later.
user_id	String	Specifies the user ID in UUID format for the ECS group. This parameter is supported in microversion 2.13 and later.

Example Request

Create an ECS group.

```
POST https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/os-server-groups
```

```
{
  "server_group": {
    "name": "test",
    "policies": ["anti-affinity"]
  }
}
```

Example Response

```
{
  "server_group": {
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "name": "test",
    "policies": [
      "anti-affinity"
    ],
    "members": [],
    "metadata": {}
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.12.2 Querying ECS Groups

Function

This API is used to query ECS groups.

URI

GET /v2.1/{project_id}/os-server-groups

[Table 6-200](#) describes the parameters in the URI.

Table 6-200 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Parameters in the following table can be used as URI parameters to filter query results.

Usage: /v2/{project_id}/os-server-groups?

Request

None

Response

[Table 6-201](#) describes the response parameters.

Table 6-201 Response parameters

Parameter	Type	Description
server_groups	Array of objects	Specifies the ECS group information. For details, see Table 6-202 .

Table 6-202 server_groups parameter information

Parameter	Type	Description
id	String	Specifies the ECS group UUID.
name	String	Specifies the ECS group name.
members	Array of strings	Specifies the ECSs in an ECS group.

Parameter	Type	Description
metadata	Object	Specifies the ECS group metadata.
project_id	String	Specifies the tenant ID in UUID format for the ECS group. This parameter is supported in microversion 2.13 and later.
policies	Array of strings	Specifies the policies associated with the ECS group. Options: <ul style="list-style-type: none">• anti-affinity: ECSs in this group must be deployed on different hosts.
user_id	String	Specifies the user ID in UUID format for the ECS group. This parameter is supported in microversion 2.13 and later.

Example Request

Query a list of ECS groups.

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/os-server-groups
```

Example Response

```
{
  "server_groups": [
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "name": "test",
      "policies": ["anti-affinity"],
      "members": [],
      "metadata": {},
      "project_id": "9c53a566cb3443ab910cf0daebca90c4"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.12.3 Querying Details About an ECS Group

Function

This API is used to query details about an ECS group.

URI

```
GET /v2.1/{project_id}/os-server-groups/{server_group_id}
```

[Table 6-203](#) describes the parameters in the URI.

Table 6-203 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group UUID.

Request

None

Response

[Table 6-204](#) describes the response parameters.

Table 6-204 Response parameters

Parameter	Type	Description
server_group	Object	Specifies the ECS group information. For details, see Table 6-205 .

Table 6-205 server_group parameters

Parameter	Type	Description
id	String	Specifies the ECS group UUID.
name	String	Specifies the ECS group name.
policies	Array of strings	Specifies the policies associated with the ECS group. <ul style="list-style-type: none">• anti-affinity: ECSs in this group must be deployed on different hosts.
members	Array of strings	Specifies the ECSs contained in the ECS group.
metadata	Object	Specifies the ECS group metadata.
project_id	String	Specifies the tenant ID in UUID format for the ECS group. This parameter is supported in microversion 2.13 and later.

Parameter	Type	Description
user_id	String	Specifies the user ID in UUID format for the ECS group. This parameter is supported in microversion 2.13 and later.

Example Request

Query details about an ECS group.

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/os-server-groups/5bbcc3c4-1da2-4437-a48a-66f15b1b13f9
```

Example Response

```
{
  "server_group": {
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "name": "test",
    "policies": ["anti-affinity"],
    "members": [],
    "metadata": {},
    "project_id": "9c53a566cb3443ab910cf0daebca90c4"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

6.12.4 Deleting an ECS Group

Function

This API is used to delete an ECS group.

URI

```
DELETE /v2.1/{project_id}/os-server-groups/{server_group_id}
```

[Table 6-206](#) describes the parameters in the URI.

Table 6-206 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group UUID.

Request

None

Response

None

Example Request

Delete a specified ECS group.

```
DELETE https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/os-server-groups/  
5bbcc3c4-1da2-4437-a48a-66f15b1b13f9
```

Returned Values

See [Returned Values for General Requests](#).

6.13 ECS Operation Management

6.13.1 Querying Operations on an ECS

Function

This API is used to query all historical operations on an ECS.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-instance-actions

[Table 6-207](#) describes the parameters in the URI.

Table 6-207 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-208](#) describes the response parameters.

Table 6-208 Response parameters

Parameter	Type	Description
instanceActions	Array of Object	Specifies operations performed on the ECS. For details, see Table 6-209 .

Table 6-209 instanceActions field description

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the action. Options: create , delete , evacuate , restore , stop , start , reboot , rebuild , revertResize , confirmResize , detach_volume , attach_volume , attach_interface , detach_interface , lock , unlock , resize , migrate , pause , unpause , rescue , unrescue , changePassword , shelve , unshelve , live-migration , trigger_crash_dump , extend_volume
instance_uuid	Yes	String	Specifies the ECS ID in UUID format.
message	Yes	String	Specifies the result status of the operation.
project_id	Yes	String	Specifies the project ID.
request_id	Yes	String	Specifies the request ID.
start_time	Yes	String	Specifies the time when the action was started.
user_id	Yes	String	Specifies the user ID.

Example Request

Query all historical operations on a specified ECS.

```
GET https://{endpoint}/v2.1/89655fe61c4c4a08b9f3e7f9095441b8/servers/e723eb40-f56e-40f9-8c8c-  
caa517fe06ba/os-instance-actions
```

Example Response

```
{  
  "instanceActions": [  
    ...  
  ]  
}
```

```
{
  "instance_uuid": "e723eb40-f56e-40f9-8c8c-caa517fe06ba",
  "user_id": "752be40780484291a9cc7ae50fff3e6d",
  "start_time": "2014-12-16T10:58:14.000000",
  "request_id": "req-ee56c2b5-d33b-4749-ae83-09281dbbb716",
  "action": "resize",
  "message": "Error",
  "project_id": "89655fe61c4c4a08b9f3e7f9095441b8"
},
{
  "instance_uuid": "e723eb40-f56e-40f9-8c8c-caa517fe06ba",
  "user_id": "752be40780484291a9cc7ae50fff3e6d",
  "start_time": "2014-12-16T10:57:56.000000",
  "request_id": "req-23cfd57f-c58a-45cd-86a6-eab3e38f3753",
  "action": "resize",
  "message": "Error",
  "project_id": "89655fe61c4c4a08b9f3e7f9095441b8"
}
]
```

Returned Values

See [Returned Values for General Requests](#).

6.13.2 Querying ECS Operations by Request ID

Function

This API is used to query a request of an ECS.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-instance-actions/{request_id}

[Table 6-210](#) describes the parameters in the URI.

Table 6-210 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
request_id	Yes	Specifies the request ID.

Request

None

Response

[Table 6-211](#) describes the response parameters.

Table 6-211 Response parameters

Parameter	Type	Description
instanceAction	Object	Specifies an operation performed on the ECS. For details, see Table 6-212 .

Table 6-212 instanceAction field description

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the action name.
instance_uuid	Yes	String	Specifies the ECS ID in UUID format.
message	Yes	String	Specifies the result status of the action.
project_id	Yes	String	Specifies the project ID.
request_id	Yes	String	Specifies the request ID.
start_time	Yes	String	Specifies the time when the action was started.
user_id	Yes	String	Specifies the user ID.
events	Yes	Array of objects	Describes events. For details, see Table 6-213 .

Table 6-213 events field description

Parameter	Mandatory	Type	Description
event	Yes	String	Specifies the action name.
result	Yes	String	Specifies the execution result.
traceback	Yes	String	Specifies the error message.
start_time	Yes	String	Specifies the time when the event was started.
finish_time	Yes	String	Specifies the time when the event was completed.

Example Request

Query a request on a specified ECS.

```
GET https://{endpoint}/v2.1/89655fe61c4c4a08b9f3e7f9095441b8/servers/e723eb40-f56e-40f9-8c8c-  
caa517fe06ba/os-instance-actions/req-5a429946-c9cc-45cc-b5bd-68864209e5c
```

Example Response

```
{  
  "instanceAction": {  
    "instance_uuid": "e723eb40-f56e-40f9-8c8c-caa517fe06ba",  
    "user_id": "752be40780484291a9cc7ae50ff3e6d",  
    "start_time": "2014-12-11T02:17:49.000000",  
    "request_id": "req-5a429946-c9cc-45cc-b5bd-68864209e5c",  
    "action": "create",  
    "message": null,  
    "project_id": "89655fe61c4c4a08b9f3e7f9095441b8",  
    "events": [  
      {  
        "finish_time": "2014-12-11T02:17:58.000000",  
        "start_time": "2014-12-11T02:17:50.000000",  
        "traceback": null,  
        "event": "compute_build_and_run_instance",  
        "result": "Success"  
      }  
    ]  
  }  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.14 ECS Console Management

6.14.1 Obtaining ECS Management Console Logs

Function

This API is used to obtain ECS management console logs.

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 6-214](#) describes the parameters in the URI.

Table 6-214 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Constraints

This API will be discarded since a version later than microversion 2.5. When using this API, set the microversion to 2.5 or earlier.

Request

[Table 6-215](#) describes the request parameters.

Table 6-215 Request parameters

Parameter	Mandatory	Type	Description
os-getConsoleOutput	Yes	Object	Obtains ECS management console logs. For details, see Table 6-216 .

Table 6-216 os-getConsoleOutput parameter description

Parameter	Mandatory	Type	Description
length	Yes	Integer	Specifies the number of request log rows. The value is greater than or equal to -1, which indicates that the output is not limited.

Response

[Table 6-217](#) describes the response parameter.

Table 6-217 Response parameter

Parameter	Type	Description
output	String	ECS console log results

Example Request

Obtain console logs of a specified ECS.

```
POST https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/47e9be4e-a7b9-471f-92d9-ffc83814e07a/action
{
  "os-getConsoleOutput" : {
    "length" : "50"
  }
}
```

Example Response

```
{
  "output": "FAKE CONSOLEOUTPUT\nANOTHER\nLAST LINE"
}
```

Returned Values

See [Returned Values for General Requests](#).

6.15 Tag Management

6.15.1 Querying All Tags of an ECS

This API is used to query all tags of an ECS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the microversion on the client.

URI

GET /v2.1/{project_id}/servers/{server_id}/tags

[Table 6-218](#) describes the parameters in the URI.

Table 6-218 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 6-219](#) describes the response parameters.

Table 6-219 Response parameters

Parameter	Type	Description
tags	Array of strings	Specifies ECS tags.

Example Request

Query all tags attached to a specified ECS.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags
```

Example Response

Example response

```
{
  "tags": ["baz=xyy", "foo", "qux"]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.15.2 Adding Tags to an ECS

This API is used to add tags to an ECS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the microversion on the client.

URI

```
PUT /v2.1/{project_id}/servers/{server_id}/tags
```

[Table 6-220](#) describes the parameters in the URI.

Table 6-220 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 6-221](#) describes the request parameters.

Table 6-221 Request parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of strings	Specifies ECS tags. A maximum of 50 tags can be configured, and each tag can contain up to 80 characters.

Response

Table 6-222 Response parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of strings	Specifies ECS tags.

Table 6-223 Reserved tag parameters

Tag Name	Description
__type_bare metal	Specifies that the server is a BMS.
__type_virtual	Specifies that the server is an ECS.

Example Request

Add tags to a specified ECS.

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags
{
  "tags": ["baz", "foo", "qux"]
}
```

Example Response

```
{
  "tags": ["baz", "foo", "qux"]
}
```

Returned Values

See [Returned Values for General Requests](#).

6.15.3 Deleting All Tags of an ECS

This API is used to delete all tags of an ECS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the microversion on the client.

URI

```
DELETE /v2.1/{project_id}/servers/{server_id}/tags
```

[Table 6-224](#) describes the parameters in the URI.

Table 6-224 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

Delete all tags from a specified ECS.

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.15.4 Adding a Tag to an ECS

This API is used to add a tag to an ECS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the microversion on the client.

Constraints

- The tag contains a maximum of 80 characters.
- A maximum of 50 tags can be added to an ECS.
- An empty tag cannot be created.

URI

```
PUT /v2.1/{project_id}/servers/{server_id}/tags/{tag}
```

[Table 6-225](#) describes the parameters in the URI.

Table 6-225 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
tag	Yes	Specifies the key of the tag to be added.

Request

None

Response

Table 6-226 Response parameters

Parameter	Type	Description
message	String	Example: " \n\n\n"
code	String	Example: "201 Created"
title	String	Example: "Created"

Example Request

Add a tag to a specified ECS.

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags/{tag}
```

Example Response

By default, the response is in HTML format.

```
<html>
<head>
  <title>201 Created</title>
</head>
<body>
  <h1>201 Created</h1>
  <br /><br />
</body>
</html>
```

JSON format

```
{
  "message": "<br /><br />\n\n\n",
```

```
"code": "201 Created",  
"title": "Created"  
}
```

Returned Values

See [Returned Values for General Requests](#).

6.15.5 Querying a Specified Tag for an ECS

This API is used to query whether an ECS has a specified tag.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the microversion on the client.

URI

GET /v2.1/{project_id}/servers/{server_id}/tags/{tag}

[Table 6-227](#) describes the parameters in the URI.

Table 6-227 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
tag	Yes	Specifies the key of the tag to be queried. If no key is specified, all tags of the ECS will be displayed.

Request

None

Response

None

Example Request

Query whether an ECS has a specified tag.

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags/{tag}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.15.6 Deleting a Specified Tag from an ECS

This API is used to delete a specified tag from an ECS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the microversion on the client.

Constraints

- The tag contains a maximum of 80 characters.
- If a tag contains non-URL-safe characters, perform URL encoding.

URI

DELETE /v2.1/{project_id}/servers/{server_id}/tags/{tag}

[Table 6-228](#) describes the parameters in the URI.

Table 6-228 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
tag	Yes	Specifies the key of the tag to be deleted. If no key is specified, all tags of the ECS will be deleted.

Request

None

Response

None

Example Request

Delete a specified tag from a specified ECS.

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags/{tag}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

6.16 Historical Versions

V2 is the historical version of native OpenStack APIs. V2.1 is recommended.

NOTE

To switch an OpenStack API from V2.1 to V2, change **2.1** in the native API URI to **2**.
The history version V2 does not support microversion functions.

7 Application Examples

7.1 Creating an ECS

Scenarios

This section describes how to create an ECS by calling APIs. For details, see [Calling APIs](#).

An ECS can be created using a disk or image. This section uses an image as an example to describe how to create an ECS.

Involved APIs

Creating an ECS involves viewing flavors and AZs as well as creating EVS disks. The following APIs are required:

- [Querying Details About ECS Flavors](#): Determine the flavor of the ECS to be created.
- [Querying Image Details](#): Determine the image based on which the ECS is to be created.
- [Querying Networks](#): Determine the network configuration of the ECS.
- [Creating and Importing an SSH Key Pair](#): Set the login mode to **Key pair**.
- [Creating an ECS](#): Create an ECS authenticated using a key pair.
- [Querying Details About an ECS](#): Verify that the ECS has been created.

Procedure

Step 1 Determine the ECS flavor.

1. View ECS flavors.
 - API

URI format: GET /v2.1/{project_id}/flavors/detail?
minDisk,minRam,is_public,sort_key,sort_dir}

The fields following the question mark (?) are optional, which are used for querying ECS flavors. For details, see [Querying Details About ECS Flavors](#).

- Example request

GET `https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/flavors/detail`

Obtain `{endpoint}` from [Regions and Endpoints](#).

- Example response

```
{
  "flavors": [
    {
      "name": "c1.2xlarge",
      "links": [
        {
          "href": "https://xxx/v2.1/74610f3a5ad941998e91f076297ecf27/flavors/c1.2xlarge",
          "rel": "self"
        },
        {
          "href": "https://xxx/74610f3a5ad941998e91f076297ecf27/flavors/c1.2xlarge",
          "rel": "bookmark"
        }
      ]
    },
    {
      "ram": 8192,
      "OS-FLV-DISABLED:disabled": false,
      "vcpus": 8,
      "swap": "",
      "os-flavor-access:is_public": true,
      "rxtx_factor": 1,
      "OS-FLV-EXT-DATA:ephemeral": 0,
      "disk": 0,
      "id": "c1.2xlarge"
    }
  ]
}
```

2. Select a flavor based on site requirements and record the flavor ID.

Step 2 Determine the image.

1. Query images.

- API

URI format: GET `/v2.1/{project_id}/images/detail`

For details, see [Querying Image Details \(Discarded\)](#).

- Example request

GET `https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/images/detail`

Obtain `{endpoint}` from [Regions and Endpoints](#).

- Example response

```
{
  "images": [
    {
      "OS-EXT-IMG-SIZE:size": 0,
      "metadata": {
        "__os_type": "Linux",
        "hw_vif_multiqueue_enabled": "true",
        "__imagetype": "gold",
        "__quick_start": "true",
        "virtual_env_type": "FusionCompute",
        "__support_xen": "true",
        "__support_kvm": "true",
        "__image_source_type": "uds",
      }
    }
  ]
}
```

```
    "__platform": "EulerOS",
    "__os_version": "EulerOS 2.2 64bit",
    "__os_bit": "64",
    "__isregistered": "false"
  },
  "created": "2018-05-14T06:13:50Z",
  "minRam": 0,
  "name": "DBS-MySQL-Image_2.1.3.3",
  "progress": 100,
  "links": [
    {
      "rel": "self",
      "href": "https://None/v2.1/74610f3a5ad941998e91f076297ecf27/images/11e8f727-d439-4ed1-b3b8-33f46c0379c4"
    },
    {
      "rel": "bookmark",
      "href": "https://None/74610f3a5ad941998e91f076297ecf27/images/11e8f727-d439-4ed1-b3b8-33f46c0379c4"
    },
    {
      "rel": "alternate",
      "href": "https://None/images/11e8f727-d439-4ed1-b3b8-33f46c0379c4",
      "type": "application/vnd.openstack.image"
    }
  ],
  "id": "11e8f727-d439-4ed1-b3b8-33f46c0379c4",
  "updated": "2018-05-14T06:13:52Z",
  "minDisk": 40,
  "status": "ACTIVE"
}
]
```

2. Select an image based on site requirements and record the image ID.

Step 3 Determine the network configuration.

1. Query networks.

- API

URI format: GET /v2.1/{project_id}/os-networks

For details, see [Querying Networks](#).

- Example request

GET https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/os-networks

Obtain {endpoint} from [Regions and Endpoints](#).

- Example response

```
{
  "networks": [
    {
      "id": "07a9557d-4256-48ae-847c-415a9c8f7ff6",
      "label": "b_tt3_td1b",
      "broadcast": null,
      "cidr": null,
      "dns1": null,
      "dns2": null,
      "gateway": null,
      "netmask": null,
      "cidr_v6": null,
      "gateway_v6": null,
      "netmask_v6": null
    }
  ]
}
```


2. Select a network based on site requirements and record the network ID.

Step 4 Set the login mode to **Key pair**.

1. Create a key pair.

- API

URI format: POST /v2.1/{project_id}/os-keypairs

For details, see [Creating and Importing an SSH Key Pair](#).

- Example request

POST https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/os-keypairs

Obtain {endpoint} from [Regions and Endpoints](#).

Body:

```
{
  "keypair": {
    "type": "ssh",
    "name": "demo1",
    "user_id": "fake"
  }
}
```

- Example response

```
{
  "keypair": {
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCrR5Gcwlh5ih7JOvzIUuQxS5qzWWPMYHeDXkDKSQ9W
5pumOV05SiO3WCswnaQ5xMdOl31mNiHtwlwq9dJi7X6jBB2shT*****
***** Generated-by-Nova\n",
    "private_key": "-----BEGIN RSA PRIVATE KEY-----
\nMII EoglBAAKCAQEAQ0eRnMJYeYoeyTr8yFLkMUuas1ljzGB3g15AykkPVuabpJld
\nOUojt1grMJ2kOcHTpd9ZjYh7cJcKvXSyu1+oyQdrlUw/tNBuVrsJAWxVOAi77d
\nQeOLtDVIkmyd+TQL1tv+F76V5vTslkNweYHumWOxLlt/FJ4fqZG4T5GMTQQivMqD\npal0IVrO
+Wm3cWQYvNdf/EcC3DYhYqHANKRsbUYwXaRenI/tU1PjnH2XUJ69ABWz\ntdc
+8sXyMoMMM1U4FLiTWzGyh0rUKkW5JXzJR2OEqT0IG+0Tf2Glyk0El0/OJPg/\ncZQzaO1o
+H8DiUzs/7Pz72yDqo0R7fQ
+mOCCn*****
*****\n-----END RSA PRIVATE KEY-----\n",
    "user_id": "f79791beca3c48159ac2553fff22e166",
    "name": "demo1",
    "fingerprint": "57:a7:a2:ed:5f:aa:e7:*.:.*.:.*.:.*.:.*.:.*"
  }
}
```

2. Import the key pair.

- API

URI format: POST /v2.1/{project_id}/os-keypairs

For details, see [Creating and Importing an SSH Key Pair](#).

- Example request

POST https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/os-keypairs

Obtain {endpoint} from [Regions and Endpoints](#).

Body:

```
{
  "keypair": {
```

```
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQBAQDY8wMTdBYiJg62o6eShoOISkx3CZ3cE6PHisDbIfK3Y0B
g7EHV7iV9c74pqsrIhK0xuGUuO1NxDQWbkwLTPN4F9Iy5CI*****
***** Generated-by-Nova\n",
    "type": "ssh",
    "name": "demo2",
    "user_id": "fake"
  }
}
```

– Example response

```
{
  "keypair": {
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQBAQDY8wMTdBYiJg62o6eShoOISkx3CZ3cE6PHisDbIfK3Y0B
g7EHV7iV9c74pqsrIhK0xuGUuO1NxDQWbkwLTPN4F9Iy5CI*****
***** Generated-by-Nova\n",
    "user_id": "f79791beca3c48159ac2553fff22e166",
    "name": "demo2",
    "fingerprint": "dd:44:45:49:d9:f6:4f:**:**:**:**:**:**:**"
  }
}
```

- Record the name in the response body, for example, **demo2**.

Step 5 Create an ECS authenticated using the key pair.

- API

URI format: POST /v2.1/{project_id}/servers

For details about API constraints and request parameters, see [Creating an ECS](#).

NOTE

In this example, the ECS is created using a specified image.

- In **block_device_mapping_v2**, set **source_type** to **image**, **uuid** to the image ID, **destination_type** to **volume**, and **boot_index** to **0**.
 - The **volume_size** must be greater than or equal to the minimum value specified in the image metadata.
- Example request

POST https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers

Obtain {endpoint} from [Regions and Endpoints](#).

Body:

```
{
  "server": {
    "flavorRef": "c1.large",
    "name": "zctestvm1",
    "block_device_mapping_v2": [{
      "source_type": "image",
      "destination_type": "volume",
      "volume_type": "SSD",
      "volume_size": "40",
      "delete_on_termination": "true",
      "uuid": "11e8f727-d439-4ed1-b3b8-33f46c0379c4",
      "boot_index": "0"
    }],
    "networks": [{
      "uuid": "fb68519f-a7c0-476e-98d4-2e4cf6de6def"
    }],
    "key_name": "demo2",
    "availability_zone": "az_test_01"
  }
}
```

```
}  
}  
• Example response  
{  
  "job_id": "ff808082739334d80173941567f21d4f",  
  "serverIds": [  
    "6d311127-bce1-48db-bf0f-cac9f8f7f077"  
  ]  
}
```

Step 6 Verify the ECS creation.

- API

URI format: GET /v2.1/{project_id}/servers/{server_id}

For details, see [Querying Details About an ECS](#).

- Example request

GET https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers/
0c71c0da-8852-4c56-a1d1-3a9b9bcb6da6

where,

0c71c0da-8852-4c56-a1d1-3a9b9bcb6da6 is the UUID of the created ECS.

Obtain {endpoint} from [Regions and Endpoints](#).

- Example response

```
{  
  "server": {  
    "tenant_id": "74610f3a5ad941998e91f076297ecf27",  
    "addresses": {  
      "2a6f4aa6-d93e-45f5-a8cb-b030dbf8cd68": [  
        {  
          "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:88:01:1b",  
          "OS-EXT-IPS:type": "fixed",  
          "addr": "192.168.2.192",  
          "version": 4  
        }  
      ]  
    },  
    "metadata": {},  
    "OS-EXT-STS:task_state": null,  
    "OS-DCF:diskConfig": "MANUAL",  
    "OS-EXT-AZ:availability_zone": "az_test_01",  
    "links": [  
      {  
        "rel": "self",  
        "href": "https://None/v2.1/74610f3a5ad941998e91f076297ecf27/servers/0c71c0da-8852-4c56-  
a1d1-3a9b9bcb6da6"  
      },  
      {  
        "rel": "bookmark",  
        "href": "https://None/74610f3a5ad941998e91f076297ecf27/servers/0c71c0da-8852-4c56-  
a1d1-3a9b9bcb6da6"  
      }  
    ],  
    "OS-EXT-STS:power_state": 1,  
    "id": "0c71c0da-8852-4c56-a1d1-3a9b9bcb6da6",  
    "os-extended-volumes:volumes_attached": [  
      {  
        "id": "b551445a-e749-4d53-932a-638a455cb6c3"  
      }  
    ],  
    "OS-EXT-SRV-ATTR:host": "pod1_test_01",  
    "image": {  
      "links": [  
        {  
          "rel": "bookmark",
```

```
    "href": "https://None/74610f3a5ad941998e91f076297ecf27/images/11e8f727-d439-4ed1-  
b3b8-33f46c0379c4"  
  },  
  ],  
  "id": "11e8f727-d439-4ed1-b3b8-33f46c0379c4"  
},  
"OS-SRV-USG:terminated_at": null,  
"accessIPv4": "",  
"accessIPv6": "",  
"created": "2018-05-25T01:47:11Z",  
"hostId": "b2792bef989888d2df1f51bff81de5ac58a4117f4e9ec3059c1a0410",  
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova001@36",  
"key_name": null,  
"flavor": {  
  "links": [  
    {  
      "rel": "bookmark",  
      "href": "https://None/74610f3a5ad941998e91f076297ecf27/flavors/c1.large"  
    }  
  ],  
  "id": "c1.large"  
},  
"security_groups": [  
  {  
    "name": "default"  
  }  
],  
"config_drive": "",  
"OS-EXT-STS:vm_state": "active",  
"OS-EXT-SRV-ATTR:instance_name": "instance-001883cd",  
"user_id": "f79791beca3c48159ac2553fff22e166",  
"name": "zttestvm1",  
"progress": 0,  
"OS-SRV-USG:launched_at": "2018-05-25T01:47:55.755922",  
"updated": "2018-05-25T01:47:55Z",  
"status": "ACTIVE"  
}  
}
```

----End

8 Data Structure

8.1 Data Structure for Creating ECSs

Contents

- [publicip Field Description](#)
- [security_groups Field Description](#)
- [eip Field Description](#)
- [bandwidth Field Description](#)
- [extendparam Field Description for Creating Disks](#)
- [extendparam Field Description for Creating ECSs](#)
- [metadata Field Description for Creating Disks](#)
- [metadata Field Description for Creating ECSs](#)
- [os:scheduler_hints Field Description](#)
- [binding:profile Field Description](#)
- [extra_dhcp_opts Field Description](#)

publicip Field Description

This field is used by the following APIs:

- [Creating ECSs /v1/{project_id}/cloudservers](#)

Table 8-1 publicip field description

Parameter	Mandator y	Type	Description
id	No	String	Specifies the ID of the existing EIP assigned to the ECS to be created. The value is in UUID format. Only EIPs in DOWN state can be assigned.

Parameter	Mandatory	Type	Description
eip	No	Object	Specifies an EIP that will be automatically assigned to an ECS. For details, see Table 8-3 .
delete_on_termination	No	Boolean	Specifies whether the EIP is released when the ECS where the EIP is bound is deleted. <ul style="list-style-type: none">• true: The EIP is released when the ECS is deleted.• false: The EIP is not released when the ECS is deleted. The default value is false .

 NOTE

Either **id** or **eip** in the **publicip** field can be configured.

security_groups Field Description

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers

Table 8-2 security_groups field description

Parameter	Mandatory	Type	Description
id	No	String	Specifies the ID of the security group to which an ECS is to be added. The configuration will take effect on the NICs of the ECS. You need to specify the ID of an existing security group in UUID format. Otherwise, the default security group will be used at the underlying layer.

eip Field Description

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers

Table 8-3 eip field description

Parameter	Mandatory	Type	Description
iptype	Yes	String	Specifies the EIP type. For details about the enumerated values, see the publicip field in "Assigning an EIP" in <i>Virtual Private Cloud API Reference</i> .
bandwidth	Yes	Object	Specifies the bandwidth of an EIP. For details, see bandwidth Field Description .

bandwidth Field Description

This field is used by the following APIs:

- Creating ECSs `/v1/{project_id}/cloudservers`

Table 8-4 bandwidth field description

Parameter	Mandatory	Type	Description
size	Yes	Integer	Specifies the bandwidth size. Specifies the bandwidth (Mbit/s). The value ranges from 1 to 1,000. The specific range may vary depending on the configuration in each region. You can see the bandwidth range of each region on the management console.
sharetype	Yes	String	Specifies the bandwidth sharing type. Enumerated values: PER (indicates exclusive bandwidth) and WHOLE (indicates shared bandwidth)
chargemode	Yes	String	Specifies the bandwidth billing mode. <ul style="list-style-type: none">• If the field value is traffic, the ECS is billed by traffic.• If the field value is others, creating the ECS will fail.

extendparam Field Description for Creating Disks

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers

Table 8-5 extendparam field description for creating disks

Parameter	Mandatory	Type	Description
snapshotId	No	String	<p>Specifies the snapshot ID or ID of the original data disk contained in the full-ECS image.</p> <p>Application scenarios: This parameter is used if an ECS is created using a full-ECS image, and the image contains one or more data disks.</p> <p>If you use a full-ECS image to create an ECS, the system automatically restores the data type and data from the data disks in the image. The snapshotId parameter allows you to specify the disk type for the original data disk after restoration.</p> <p>NOTE</p> <ul style="list-style-type: none"> You are advised to specify snapshotId for each original data disk. If you are required to change a disk size, ensure that the changed disk size is greater than or equal to the size of the original data disk. Otherwise, restoring data of the original data disk will fail. To set disk sharing, you need to specify the sharing attribute. To set disk encryption, you need to specify the encryption attribute in the metadata field. <p>Working rules: snapshotId uniquely identifies an original data disk contained in a full-ECS image. You can use snapshotId to obtain the information of the original data disk for data restoration.</p> <p>Obtaining snapshotId through the management console: Log in to the management console, choose Elastic Volume Service > Snapshot. Then, use the name of the original data disk to find the snapshot ID or the original disk ID.</p> <p>Obtaining snapshotId through the API:</p>

Parameter	Mandatory	Type	Description
			<p>If you have obtained the full-ECS image ID, obtain the Cloud Backup and Recovery (CBR) or Cloud Server Backup Service (CSBS) backup ID associated with the full-ECS image ID by following the instructions provided in the API for querying image details.</p> <ul style="list-style-type: none"> • If CBR backup is used, use the CBR backup ID to obtain the backup. The resource_id or snapshot_id contained in the children field in the response is the desired snapshotId. For details, see the API for "Querying a Specified Backup" in <i>Cloud Backup and Recovery User Guide</i>. • If CSBS backup is used, use the CSBS backup ID to obtain the backup. The source_volume_id or snapshot_id contained in the volume_backups field in the response is the desired snapshotId. For details, see the API for "Querying a Single Backup" in <i>Cloud Server Backup Service User Guide</i>.

extendparam Field Description for Creating ECSs

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers

Table 8-6 extendparam field description for creating ECSs (for V1 APIs)

Parameter	Mandatory	Type	Description
regionID	No	String	Specifies the ID of the region where the ECS resides.

metadata Field Description for Creating Disks

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers

Table 8-7 metadata field description for creating disks

Parameter	Mandatory	Type	Description
__system__encrypted	No	String	Specifies encryption in metadata . The value can be 0 (encryption disabled) or 1 (encryption enabled). If this parameter does not exist, the disk will not be encrypted by default.
__system__cmkid	No	String	Specifies the CMK ID, which indicates encryption in metadata . This parameter is used with __system__encrypted . NOTE For details about how to obtain the CMK ID, see "Querying the List of CMKs" in <i>Key Management Service API Reference</i> .

metadata Field Description for Creating ECSs

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers

Table 8-8 metadata reserved field description

Parameter	Mandatory	Type	Description
admin_pass	No	String	Specifies the password of user Administrator for logging in to a Windows ECS. For details, see Function . NOTE This parameter is mandatory when a Windows ECS using password authentication is created.
op_svc_userid	No	String	Specifies the user ID. NOTE The value of this parameter is that of IAM User ID of the current login account. If you log in as an IAM user, obtain the IAM user ID of that IAM user.

Parameter	Mandatory	Type	Description
agency_name	No	String	Specifies the IAM agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for ECSs to access cloud services.
BYOL	No	String	If you have an OS or a software license (a license certified based on the number of physical servers and cores), you can migrate your services to the cloud platform in BYOL mode to continue using your existing licenses. <ul style="list-style-type: none"> • True: Use your existing licenses. • False: System licenses are used.

os:scheduler_hints Field Description

This field is used by the following APIs:

- Creating ECSs: /v1/{project_id}/cloudservers
- Creating ECSs (native): /v2.1/{project_id}/servers

Table 8-9 os:scheduler_hints field description (request parameters)

Parameter	Mandatory	Type	Description
group	No	String	Specifies the ECS group ID in UUID format. Obtain the parameter value from the console or by referring to Querying ECS Groups . NOTE Ensure that the ECS group uses the anti-affinity policy.
tenancy	No	String	Creates ECSs on a dedicated or shared host. The value of this parameter can be dedicated or shared .
dedicated_host_id	No	String	Specifies the dedicated host ID. NOTE A DeH ID takes effect only when tenancy is set to dedicated .

Table 8-10 os:scheduler_hints field description (response parameters)

Parameter	Type	Description
group	Array of strings	Specifies the ECS group ID in UUID format. Obtain the parameter value from the console or by referring to Querying ECS Groups .
tenancy	Array of strings	Creates ECSs on a dedicated or shared host. The value of this parameter can be dedicated or shared .
dedicated_host_id	Array of strings	Specifies the dedicated host ID. NOTE A DeH ID takes effect only when tenancy is set to dedicated .

binding:profile Field Description

This field is used by the following APIs:

- Creating ECSs: /v1/{project_id}/cloudservers

Table 8-11 binding:profile field description

Parameter	Mandatory	Type	Description
disable_security_groups	No	Boolean	Specifies whether the NIC of an ECS for HANA is not added to a security group. NOTE <ul style="list-style-type: none"> • The primary NIC of the ECS for HANA must be added to a security group. • At most one NIC of the HANA ECS is allowed not to be added to any security groups.

extra_dhcp_opts Field Description

This field is used by the following APIs:

- Creating ECSs: /v1/{project_id}/cloudservers

Table 8-12 extra_dhcp_opts field description

Parameter	Mandatory	Type	Description
opt_value	Yes	Integer	Specifies the NIC MTU, which ranges from 1280 to 8888.
opt_name	Yes	String	Set the parameter value to 26 .

8.2 Data Structure for Querying Details About ECSs

Table 8-13 address parameters

Parameter	Type	Description
version	String	Specifies the IP address version. <ul style="list-style-type: none">• 4: indicates IPv4.• 6: indicates IPv6.
addr	String	Specifies the IP address.
OS-EXT-IPS:type	String	Specifies the IP address type. <ul style="list-style-type: none">• fixed: indicates the private IP address.• floating: indicates the floating IP address.
OS-EXT-IPS-MAC:mac_addr	String	Specifies the MAC address.
OS-EXT-IPS:port_id	String	Specifies the port ID corresponding to the IP address.

Table 8-14 flavor parameters

Parameter	Type	Description
id	String	Specifies the ECS flavor ID.
name	String	Specifies the ECS flavor name.
disk	String	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. The field is invalid in this system.
vcpus	String	Specifies the number of vCPUs in the ECS flavor.

Parameter	Type	Description
ram	String	Specifies the memory size (MB) in the ECS flavor.

Table 8-15 security_groups parameters

Parameter	Type	Description
name	String	Specifies the security group name or UUID.
id	String	Specifies the security group ID.

Table 8-16 os-extended-volumes:volumes_attached parameters

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.
delete_on_termination	String	Specifies whether the disk is deleted with the ECS. <ul style="list-style-type: none">● true: indicates that the disk is deleted with the ECS.● false: indicates that the disk is not deleted with the ECS. This parameter is supported in microversion 2.3 and later.
bootIndex	String	Specifies the EVS disk boot sequence. <ul style="list-style-type: none">● 0 indicates the system disk.● Non-0 indicates a data disk.
device	String	Specifies the drive letter of the EVS disk, which is the device name of the EVS disk.

Table 8-17 metadata parameters

Parameter	Type	Description
charging_mode	String	Specifies the ECS billing mode.
vpc_id	String	Specifies the ID of the VPC where the ECS is located.

Parameter	Type	Description
EcmResStatus	String	Specifies the ECS frozen status. <ul style="list-style-type: none">• normal: The ECS is not frozen.• freeze: The ECS has been frozen. NOTE The system automatically adds this field, which is mandatory, after an ECS is frozen or unfrozen.
metering.image_id	String	Specifies the image ID of the ECS.
metering.imagetype	String	Specifies the image type. The following types are supported: <ul style="list-style-type: none">• Public image (gold)• Private image (private)• Shared image (shared)
metering.resourcespeccode	String	Specifies the resource specifications of the ECS.
metering.resourcetype	String	Specifies the resource type of the ECS. Value 1 indicates ECSs.
cascaded.instance_extainfo	String	Specifies the extended information about the internal ECSs.
image_name	String	Specifies the image name of the ECS.
agency_name	String	Specifies the IAM agency name. An agency is created by a tenant administrator on IAM to provide temporary credentials for ECSs to access cloud services.
os_bit	String	Specifies the number of bits in the operating system: 32 or 64 .
os_type	String	Specifies the OS type. The value can be Linux or Windows .
lockCheckEndpoint	String	Specifies the callback URL for checking whether ECS locking is enabled. <ul style="list-style-type: none">• If ECS locking is enabled, the ECS is locked.• If ECS locking is disabled, the ECS is unlocked, and invalid locks are deleted.

Parameter	Type	Description
lockSource	String	Specifies the lock source. <ul style="list-style-type: none">• Order lock (ORDER)
lockSourceId	String	Specifies the ECS lock source ID. If lockSource is set to ORDER , lockSourceId is the order ID.
lockScene	String	Specifies the ECS lock type.
virtual_env_type	String	<ul style="list-style-type: none">• If an ECS is created using an iOS image, the value of this parameter is Isolmage.• If an ECS is created using a non-iOS image, the value of this parameter is FusionCompute in versions earlier than 19.5.0, and this parameter will be unavailable in versions later than 19.5.0. NOTE <ul style="list-style-type: none">• The virtual_env_type cannot be added, deleted, or modified.

Table 8-18 sys_tags parameters

Parameter	Type	Description
key	String	Specifies the system tag key.
value	String	Specifies the system tag value.

Table 8-19 image parameters

Parameter	Type	Description
id	String	Specifies the image ID.

8.3 Data Structure for Querying Details About Specifications

os_extra_specs (flavor) Field Description

This field is used by the following APIs:

- Querying details about flavors and extended flavor information: `/v1/{project_id}/cloudservers/flavors`

- Querying details about the extended ECS flavor field: /v1/{project_id}/flavors/{flavor_id}/os-extra_specs

Table 8-20 os_extra_specs field description

Parameter	Type	Description
ecs:performance_type	String	Specifies the ECS flavor type: <ul style="list-style-type: none">• normal: general computing• computingv3: general computing-plus• highmem: memory-optimized• cpuv1: computing I• cpuv2: computing II• highcpu: high-performance computing• diskintensive: disk-intensive• gpu: GPU-accelerated• saphana: large-memory
resource_type	String	Specifies the resource type. resource_type is used to differentiate between the types of the physical servers accommodating ECSs.
instance_vnic:type	String	Specifies the NIC type. The value of this parameter is consistently enhanced , indicating that network enhancement ECSs are to be created.
instance_vnic:instance_bandwidth	String	Specifies the maximum bandwidth in the unit of Mbit/s. The maximum value of this parameter is 10,000 .
instance_vnic:max_count	String	Specifies the maximum number of NICs. The maximum value of this parameter is 4.

Parameter	Type	Description
quota:local_disk	String	<p>The value of this parameter is in format of {type}:{count}:{size}:{safeFormat}.</p> <ul style="list-style-type: none">● type: indicates the disk type, which can only be HDD.● count: indicates the number of local disks. The following types are supported:<ul style="list-style-type: none">- For D1 ECSs, the value can be 3, 6, 12, or 24.- For D2 ECSs, the value can be 2, 4, 8, 12, 16, or 24.- For D3 ECSs, the value can be 2, 4, 8, 12, 16, 24, or 28.● size: indicates the capacity of a single disk, in GB. Currently, only 1675 is supported. The actual disk size is 1800, and the available size after formatting is 1675.● safeFormat: indicates whether the local disks of the ECS are securely formatted. The following types are supported:<ul style="list-style-type: none">- For D1 ECSs, the value is FALSE.- For D2 or D3 ECSs, the value is True. <p>NOTE This field is dedicated for disk-intensive ECSs.</p>
quota:nvme_ssd	String	<p>The value of this parameter is in the format of {type}:{spec}:{num}:{size}:{safeFormat}.</p> <ul style="list-style-type: none">● type: indicates the capacity of a single NVME SSD disk attached to the ECS, which can only be 1.6 TB or 3.2 TB.● spec: indicates the specifications of the NVMe SSD disk, which can be large (large specifications) or lvs (small specifications). If spec is set to large, only I series (for example, I3) is supported.● num indicates the number of local disks.● size: indicates the capacity, in the unit of GB, of the disk used by the guest user. If the spec value is large, the value of this parameter is the size of a single disk attached to the ECS. If the value of spec is lvs, the value of size is an integer multiple of 50.● safeFormat: indicates whether the local disks of the ECS are securely formatted. If safeFormat is set to True, only I series (for example, I3) is supported. <p>NOTE This field is dedicated for ultra-high I/O ECSs.</p>

Parameter	Type	Description
extra_spec:io:persistent_grant	String	Specifies whether persistence is supported. The value of this parameter is true . This parameter indicates that the ECS is persistently authorized to access the storage. NOTE This field is dedicated for disk-intensive D1 ECSs.
ecs:generation	String	Specifies the generation of an ECS type. For example, 3 in s3 indicates the general-purpose third-generation ECSs. For details about flavors and generations, see "ECS Specifications" in the <i>Elastic Cloud Server User Guide</i> .
ecs:virtualization_env_types	String	Specifies a virtualization type. <ul style="list-style-type: none">If the parameter value is FusionCompute, the ECS uses Xen virtualization.If the parameter value is CloudCompute, the ECS uses KVM virtualization. NOTE This field is optional.
pci_passthrough:enable_gpu	String	Specifies whether the GPU is passthrough. If the value is true , the GPU is passthrough.
pci_passthrough:gpu_specs	String	Specifies the technology used on the G1 and G2 ECSs, including GPU virtualization and GPU passthrough. <ul style="list-style-type: none">If the ECS with specified specifications uses GPU virtualization and the GPU model is M60-1Q, the parameter value can be m60_1q:virt:1.If the ECS with specified specifications uses GPU passthrough and the GPU model is M60, the parameter value can be m60:direct_graphics:1.

Parameter	Type	Description
cond:operation:status	String	<p>This parameter takes effect region-wide. If an AZ is not configured in the cond:operation:az parameter, the value of this parameter is used by default. If this parameter is not set or used, the meaning of normal applies. Options:</p> <ul style="list-style-type: none"> • normal: indicates normal commercial use of the flavor. • abandon: indicates that the flavor has been taken offline (not displayed). • sellout: indicates that the flavor has been sold out. • obt: indicates that the flavor is under open beta testing (OBT). • obt_sellout: indicates that the OBT resources are sold out. • promotion: indicates that the flavor is recommended (for commercial use, which is similar to normal).
cond:operation:az	String	<p>This parameter takes effect AZ-wide. If an AZ is not configured in this parameter, the value of the cond:operation:status parameter is used by default. This parameter is in the format of "az(xx)". The value in parentheses is the flavor status in an AZ. If the parentheses are left blank, the configuration is invalid. The cond:operation:az options are the same as the cond:operation:status options.</p> <p>For example, a flavor is for commercial use in AZs 0 and 3, sold out in AZ 1, for OBT in AZ 2, and is canceled in other AZs. Then, set parameters as follows:</p> <ul style="list-style-type: none"> • cond:operation:status: abandon • cond:operation:az: az0(normal), az1(sellout), az2(obt), az3(normal) <p>NOTE Configure this parameter if the flavor status in an AZ is different from the cond:operation:status value.</p>
quota:max_rate	String	<p>Specifies the maximum bandwidth.</p> <ul style="list-style-type: none"> • Unit: Mbit/s. If a bandwidth is in the unit of Gbit/s, it must be divided by 1,000.
quota:min_rate	String	<p>Specifies the assured bandwidth.</p> <ul style="list-style-type: none"> • Unit: Mbit/s. If a bandwidth is in the unit of Gbit/s, it must be divided by 1,000.
quota:max_pps	String	<p>Specifies the maximum intranet PPS.</p> <ul style="list-style-type: none"> • Unit: number. If a value is in the unit of 10,000, it must be divided by 10,000.

Parameter	Type	Description
cond:operation:charge	String	Specifies a billing type. <ul style="list-style-type: none">All the billing types are supported if this parameter is not set.
cond:compute	String	Specifies computing constraints. <ul style="list-style-type: none">autorecovery: indicates that automatic recovery is supported.If this parameter does not exist, automatic recovery is not supported.

9 Permissions and Supported Actions

9.1 Introduction

You can use Identity and Access Management (IAM) for fine-grained permissions management of your ECSs. If your account does not need individual IAM users, you can skip this section.

New IAM users do not have any permissions assigned by default. You need to first add them to one or more groups and attach policies or roles to these groups. The users then inherit permissions from the groups and can perform specified operations on cloud services based on the permissions they have been assigned.

You can grant users permissions by using roles and policies. Roles are provided by IAM to define service-based permissions that match users' job 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.

For more information about system-defined policies supported by ECS, see "Permissions Management" in the *Elastic Cloud Server User Guide*.

NOTE

If you want to allow or deny the access to an API, use policy-based authorization.

Each 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 who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user wants to query ECSs using an API, the user must have been granted permissions that allow the **ecs:servers:list** action.

Supported Actions

ECS provides system-defined policies that can be directly used in IAM. You 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 on specified resources under specific conditions.
- APIs: REST APIs that can be called by a custom policy
- Actions: specific operations that are allowed or denied in a custom policy
- Dependencies: actions which a specific action depends on. When allowing an action for a user, you also need to allow any existing action dependencies for that user.
- IAM projects: Authorization scope of custom policies. Policies that contain actions only for IAM projects can be assigned to user groups and be applied only in IAM.

 **NOTE**

√: supported; x: not supported

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

- [Lifecycle Management](#)
- [ECS Status Management](#)
- [Batch Operations](#)
- [Network Management](#)
- [Image Management](#)
- [Security Group Management](#)
- [Specifications Query](#)
- [NIC Management](#)
- [Disk Management](#)
- [Metadata Management](#)
- [Tenant Quota Management](#)
- [SSH Key Management](#)
- [Floating IP Address Management](#)
- [ECS Group Management](#)
- [Tag Management](#)

9.2 Actions Supported by Policy-based Authorization

This section describes the actions supported by ECS in policy-based authorization.

Supported Actions

ECS provides system-defined policies that can be directly used in IAM. You 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 in a custom policy

- **Dependencies:** actions which a specific action depends on. When allowing an action for a user, you also need to allow any existing action dependencies for that user.
- **IAM projects/Enterprise projects:** the authorization scope of a custom policy. A custom policy can be applied to IAM projects or enterprise projects or both. Policies that contain actions for both IAM and enterprise projects can be used and applied for both IAM and Enterprise Management. Policies that contain actions only for IAM projects can be used and applied to IAM only. Administrators can check whether an action supports IAM projects or enterprise projects in the action list.

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

- [Lifecycle Management](#)
- [ECS Status Management](#)
- [Batch Operations](#)
- [Network Management](#)
- [Image Management](#)
- [Security Group Management](#)
- [Specifications Query](#)
- [NIC Management](#)
- [Disk Management](#)
- [Metadata Management](#)
- [Tenant Quota Management](#)
- [SSH Key Management](#)
- [Floating IP Address Management](#)
- [ECS Group Management](#)
- [Tag Management](#)

9.2.1 Lifecycle Management

Permission	API	Action	Dependencies
Creating ECSs (pay-per-use)	POST /v1/{project_id}/cloudservers	<ul style="list-style-type: none"> • Assigning a New EIP ecs:cloudServers:create • Using an Existing EIP ecs:cloudServers:create 	<ul style="list-style-type: none"> • Assigning a New EIP vpc:publicIps:create • Using an Existing EIP vpc:publicIps:update
Deleting ECSs	POST /v1/{project_id}/cloudservers/delete	ecs:cloudServers:delete	-

Permission	API	Action	Dependencies
Querying details about ECSs (native OpenStack API)	GET /v2.1/{project_id}/servers/detail	ecs:servers:list	ecs:servers:get ecs:serverVolumes:use ecs:diskConfigs:use ecs:securityGroups:use ecs:serverKeypairs:get vpc:securityGroups:get vpc:securityGroupRules:get vpc:networks:get vpc:subnets:get vpc:ports:get vpc:routers:get
Querying a list of ECSs (native OpenStack API)	GET /v2.1/{project_id}/servers	ecs:servers:list	-
Querying details about a specific ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}	ecs:servers:get	ecs:serverVolumes:use ecs:diskConfigs:use ecs:securityGroups:use ecs:serverKeypairs:get vpc:securityGroups:get vpc:securityGroupRules:get vpc:networks:get vpc:subnets:get vpc:ports:get vpc:routers:get

Permission	API	Action	Dependencies
Creating an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers POST /v2.1/{project_id}/os-volumes_boot	ecs:servers:create	ecs:servers:get ecs:serverInterfaces:use ecs:serverInterfaces:get ecs:flavors:get ecs:securityGroups:use evs:volumes:list evs:volumes:get evs:volumes:create evs:volumes:attach evs:volumes:manage vpc:securityGroups:get vpc:networks:get vpc:networks:update vpc:subnets:get vpc:subnets:update vpc:ports:create vpc:ports:update vpc:ports:get vpc:ports:delete vpc:networks:create vpc:subnets:create vpc:routers:get vpc:routers:update ims:images:list ims:images:get
Deleting an ECS (native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}	ecs:servers:delete	-

Permission	API	Action	Dependencies
Modifying an ECS (native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}	ecs:servers:update	ecs:servers:get

9.2.2 ECS Status Management

Permission	API	Action	Dependencies
Changing an ECS OS	POST /v2/{project_id}/cloudservers/{server_id}/changeos	ecs:cloudServers:changeOS	-
Reinstalling an ECS OS	POST /v2/{project_id}/cloudservers/{server_id}/reinstallos	ecs:cloudServers:rebuild	-
Modifying ECS specifications (pay-per-use)	POST /v1/{project_id}/cloudservers/{server_id}/resize	ecs:cloudServers:resize	-
Starting an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:start	ecs:servers:list
Stopping an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:stop	ecs:servers:list
Restarting an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:reboot	ecs:servers:list

Permission	API	Action	Dependencies
Modifying ECS specifications (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:resize	ecs:servers:list ecs:flavors:get ims:images:get evs:volumes:list evs:volumes:create evs:volumes:get evs:volumes:attach evs:volumes:detach evs:volumes:manage vpc:ports:get vpc:ports:update vpc:ports:create vpc:ports:delete
Rebuilding an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:rebuild	ecs:servers:list ecs:servers:update ims:images:get ims:images:list ims:images:update
Locking an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:lock	ecs:servers:list
Unlocking an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:unlock	ecs:servers:list

9.2.3 Batch Operations

Permission	API	Action	Dependencies
Stopping ECSs in a batch	POST /v1/{project_id}/cloudservers/action	ecs:cloudServers:stop	-
Restarting ECSs in a batch	POST /v1/{project_id}/cloudservers/action	ecs:cloudServers:reboot	-
Starting ECSs in a batch	POST /v1/{project_id}/cloudservers/action	ecs:cloudServers:start	-

9.2.4 Network Management

Permission	API	Action	Dependencies
Querying networks (native OpenStack API)	GET /v2.1/{project_id}/os-networks	ecs:networks:list	vpc:networks:get

9.2.5 Image Management

Permission	API	Action	Dependencies
Creating an image (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:creatimage	evs:volumes:get evs:snapshots:create ims:images:create ims:images:get ims:images:list ims:images:update ims:images:delete

9.2.6 Security Group Management

Permission	API	Action	Dependencies
Creating a security group (native OpenStack API)	POST /v2.1/{project_id}/os-security-groups	ecs:securityGroups:use	vpc:securityGroups:get vpc:securityGroups:create vpc:securityGroups:update
Deleting a security group (native OpenStack API)	DELETE /v2.1/{project_id}/os-security-groups/{security_group_id}	ecs:securityGroups:use	vpc:securityGroups:get vpc:securityGroups:delete vpc:securityGroups:update
Querying details about a security group (native OpenStack API)	GET /v2.1/{project_id}/os-security-groups/{security_group_id}	ecs:securityGroups:use	vpc:securityGroups:get
Querying security groups (native OpenStack API))	GET /v2.1/{project_id}/os-security-groups	ecs:securityGroups:use	vpc:securityGroups:get
Creating a security group rule (native OpenStack API)	POST /v2.1/{project_id}/os-security-group-rules	ecs:securityGroups:use	vpc:securityGroups:get vpc:securityGroups:update vpc:securityGroupRules:get vpc:securityGroupRules:create
Deleting a security group rule (native OpenStack API)	DELETE /v2.1/{project_id}/os-security-group-rules/{security_group_rule_id}	ecs:securityGroups:use	vpc:securityGroups:get vpc:securityGroups:update vpc:securityGroupRules:get vpc:securityGroupRules:delete

Permission	API	Action	Dependencies
Updating a security group (native OpenStack API)	PUT /v2.1/{project_id}/os-security-groups/{security_group_id}	ecs:securityGroups:use	vpc:securityGroups:get vpc:securityGroups:update
Querying security groups of a specified ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-security-groups	ecs:securityGroups:use	vpc:securityGroups:get
Adding an ECS to a security group (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:securityGroups:use	ecs:servers:get ecs:servers:list ecs:serverVolumes:use ecs:diskConfigs:use ecs:serverKeypairs:get vpc:securityGroups:get vpc:securityGroups:create vpc:securityGroups:update vpc:securityGroupRules:get vpc:networks:get vpc:subnets:get vpc:routers:get vpc:ports:get vpc:ports:update

Permission	API	Action	Dependencies
Removing a security group (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:securityGroups:use	ecs:servers:get ecs:servers:list ecs:serverVolumes:use ecs:diskConfigs:use ecs:serverKeypairs:get vpc:securityGroups:get vpc:securityGroups:delete vpc:securityGroups:update vpc:securityGroupRules:get vpc:networks:get vpc:subnets:get vpc:routers:get vpc:ports:get vpc:ports:update

9.2.7 Specifications Query

Permission	API	Action	Dependencies
Querying details about flavors and extended flavor information	GET /v1/{project_id}/cloudservers/flavors	ecs:cloudServerFlavors:get	-

Permission	API	Action	Dependencies
Querying the extra_specs value for an ECS (native OpenStack API)	GET /v2.1/{project_id}/flavors/{flavors_id}/os-extra_specs	ecs:flavors:get	-

9.2.8 NIC Management

Permission	API	Action	Dependencies
Deleting NICs from an ECS in a batch	POST /v1/{project_id}/cloudservers/{server_id}/nics/delete	ecs:cloudServerNics:delete	-
Adding NICs to an ECS in a batch	POST /v1/{project_id}/cloudservers/{server_id}/nics	ecs:cloudServers:addNics	-

Permission	API	Action	Dependencies
Adding a NIC to an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/os-interface	ecs:serverInterfaces:use	ecs:servers:get ecs:serverInterfaces:get vpc:networks:get vpc:networks:update vpc:subnets:get vpc:subnets:update vpc:ports:create vpc:ports:update vpc:ports:get vpc:networks:create vpc:subnets:create vpc:routers:get vpc:routers:update

Permission	API	Action	Dependencies
Deleting a NIC from an ECS (native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/os-interface/{id}	ecs:serverInterfaces:use	ecs:serverInterfaces:get ecs:servers:get vpc:networks:create vpc:subnets:create vpc:networks:get vpc:networks:update vpc:subnets:get vpc:subnets:update vpc:ports:delete vpc:ports:update vpc:ports:get vpc:routers:get vpc:routers:update
Querying NICs of an ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-interface	ecs:serverInterfaces:get	vpc:ports:get
Querying details about a specified NIC of an ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-interface/{id}	ecs:serverInterfaces:get	vpc:ports:get

9.2.9 Disk Management

Permission	API	Action	Dependencies
Detaching a disk from an ECS	DELETE /v1/{project_id}/cloudservers/{server_id}/detachvolume/{volume_id}	ecs:cloudServers:detachVolume	-
Attaching a disk to an ECS	POST /v1/{project_id}/cloudservers/{server_id}/attachvolume	ecs:cloudServers:attach	-
Attaching a disk to an ECS (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/os-volume_attachments	ecs:serverVolumeAttachments:create	ecs:servers:get ecs:flavors:get ecs:serverVolumes:use evs:volumes:list evs:volumes:get evs:volumes:update evs:volumes:attach evs:volumes:manage
Detaching a disk from an ECS (native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}	ecs:serverVolumeAttachments:delete	ecs:serverVolumes:use evs:volumes:list evs:volumes:get evs:volumes:update evs:volumes:detach evs:volumes:manage
Querying the disks attached to an ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments	ecs:serverVolumeAttachments:list	ecs:serverVolumes:use ecs:servers:get

Permission	API	Action	Dependencies
Querying a disk attached to an ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}	ecs:serverVolumeAttachments:get	ecs:serverVolumes:use
Creating a disk (native OpenStack API)	POST /v2.1/{project_id}/os-volumes	ecs:serverVolumes:use	evs:volumes:create
Deleting a disk (native OpenStack API)	DELETE /v2.1/{project_id}/os-volumes/{volume_id}	ecs:serverVolumes:use	evs:volumes:get evs:volumes:delete
Querying information about a disk (native OpenStack API)	GET /v2.1/{project_id}/os-volumes/{volume_id}	ecs:serverVolumes:use	evs:volumes:get
Querying brief information about disks (native OpenStack API)	GET /v2.1/{project_id}/os-volumes	ecs:serverVolumes:use	evs:volumes:get evs:volumes:list
Querying detailed information about disks (native OpenStack API)	GET /v2.1/{project_id}/os-volumes/detail	ecs:serverVolumes:use	evs:volumes:get evs:volumes:list

9.2.10 Metadata Management

Permission	API	Action	Dependencies
Querying ECS metadata (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/metadata	ecs:servers:listMetadata	-
Obtaining ECS metadata with a specified key (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/metadata/{key}	ecs:servers:getMetadata	ecs:servers:get
Deleting specified ECS metadata (native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/metadata/{key}	ecs:servers:setMetadata	-
Modifying ECS metadata with a specified key (native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}/metadata/{key}	ecs:servers:setMetadata	-
Updating ECS metadata (native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/metadata	ecs:servers:setMetadata	-
Configuring ECS metadata (native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}/metadata	ecs:servers:setMetadata	ecs:servers:get

9.2.11 Tenant Quota Management

Permission	API	Action	Dependencies
Querying quotas of a tenant	GET /v1/{project_id}/cloudservers/limits	ecs:cloudServerQuotas:get	-
Querying quotas of a tenant (native OpenStack API)	GET /v2.1/{project_id}/os-quota-sets/{project_id}?user_id={user_id}	ecs:quotas:get	-
Querying default quotas (native OpenStack API)	GET /v2.1/{project_id}/os-quota-sets/{project_id}/defaults	ecs:quotas:get	-

9.2.12 SSH Key Management

Permission	API	Action	Dependencies
Creating and importing an SSH key pair (native OpenStack API)	POST /v2.1/{project_id}/os-keypairs	ecs:serverKeypairs:create	-
Querying an SSH key pair (native OpenStack API)	GET /v2.1/{project_id}/os-keypairs/{keypair_name}	ecs:serverKeypairs:get	-

Permission	API	Action	Dependencies
Querying SSH key pairs (native OpenStack API)	GET /v2.1/{project_id}/os-keypairs	ecs:serverKeypairs:list	-
Deleting an SSH key pair (native OpenStack API)	DELETE /v2.1/{project_id}/os-keypairs/{keypair_name}	ecs:serverKeypairs:delete	-

9.2.13 Password Management

Permission	API	Action	Dependencies
Obtaining the password for logging in to a Windows ECS (native OpenStack)	GET /v2.1/{project_id}/servers/{server_id}/os-server-password	ecs:serverPasswords:manage	-
Deleting the password for logging in to a Windows ECS (native OpenStack)	DELETE /v2.1/{project_id}/servers/{server_id}/os-server-password	ecs:serverPasswords:manage	-

9.2.14 Floating IP Address Management

Permission	API	Action	Dependencies
Allocating a floating IP address (native OpenStack API)	POST /v2.1/{project_id}/os-floating-ips	ecs:serverFloatingIps:use	vpc:floatingIps:get vpc:floatingIps:create vpc:floatingIps:update vpc:ports:get
Querying floating IP addresses (native OpenStack API)	GET /v2.1/{project_id}/os-floating-ips	ecs:serverFloatingIps:use	vpc:floatingIps:get vpc:ports:get
Querying details about a floating IP address (native OpenStack API)	GET /v2.1/{project_id}/os-floating-ips/{floating_ip_id}	ecs:serverFloatingIps:use	vpc:floatingIps:get vpc:ports:get
Releasing a floating IP address (native OpenStack API)	DELETE /v2.1/{project_id}/os-floating-ips/{floating_ip_id}	ecs:serverFloatingIps:use	vpc:floatingIps:get vpc:floatingIps:delete vpc:floatingIps:update vpc:ports:get

9.2.15 ECS Group Management

Permission	API	Action	Dependencies
Deleting an ECS group	DELETE /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}	ecs:cloudServers:delete	-
Creating an ECS group	POST /v1{project_id}/cloudservers/os-server-groups	ecs:cloudServers:create	-
Adding an ECS to an ECS group	POST /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}/action	ecs:cloudServers:create	-
Removing an ECS from an ECS group	POST /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}/action	ecs:cloudServers:delete	-
Querying ECS groups	GET /v1/{project_id}/cloudservers/os-server-groups	ecs:cloudServers:list	-
Querying details about an ECS group	GET /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}	ecs:cloudServers:get	-
Creating an ECS group (native OpenStack API)	POST /v2.1/{project_id}/os-server-groups	ecs:serverGroups:manage	-

Permission	API	Action	Dependencies
Querying ECS groups (native OpenStack API)	GET /v2.1/{project_id}/os-server-groups	ecs:serverGroups:manage	-
Querying details about an ECS group (native OpenStack API)	GET /v2.1/{project_id}/os-server-groups/{server_group_id}	ecs:serverGroups:manage	-
Deleting an ECS group (native OpenStack API)	DELETE /v2.1/{project_id}/os-server-groups/{server_group_id}	ecs:serverGroups:manage	-

9.2.16 Tag Management

Permission	API	Action	Dependencies
Adding tags to an ECS in a batch Deleting tags from an ECS in a batch	POST /v1/{project_id}/servers/{server_id}/tags/action	ecs:servers:setTags	-

Permission	API	Action	Dependencies
Querying ECSs by tag	POST /v1/{project_id}/servers/resource_instances/action	ecs:servers:getTags	-
Querying tags of a specified ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/tags	ecs:servers:getTags	ecs:servers:get
Adding a tag to an ECS (native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}/tags/{tag}	ecs:servers:setTags	ecs:servers:get
Adding tags to an ECS (native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}/tags	ecs:servers:setTags	ecs:servers:get
Deleting a specified tag from an ECS (native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/tags/{tag}	ecs:servers:setTags	ecs:servers:get

Permission	API	Action	Dependencies
Querying a specified tag for an ECS (native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/tags/{tag}	ecs:servers:getTags	ecs:servers:get
Deleting tags from an ECS (native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/tags	ecs:servers:setTags	ecs:servers:get

10 Common Parameters

10.1 Returned Values for General Requests

- Normal

Returned Value	Description
200	Request succeeded.
201	Request processed.
202	After the task is successfully delivered, the task to be delivered shall be postponed because the system is busy.
204	Task delivered.

- Abnormal

Returned Value	Description
300 multiple choices	The requested resource has multiple available responses.
400 Bad Request	The server failed to process the request.
401 Unauthorized	You need to enter the username and password to access the page requested.
403 Forbidden	You are forbidden to access the page requested.
404 Not Found	The server cannot find the page requested.
405 Method Not Allowed	You are not allowed to use the method specified in the request.

Returned Value	Description
406 Not Acceptable	The response generated by the server cannot be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then, the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request cannot be processed due to a conflict.
429 Too Many Requests	The request throttling threshold is reached.
500 Internal Server Error	Failed to complete the request because an internal service error occurs. A service exception occurred.
501 Not Implemented	Failed to complete the request because an internal service error occurs. The server does not support the requested function.
502 Bad Gateway	Failed to complete the request because an internal service error occurs. Failed to complete the request because the server receives an invalid request.
503 Service Unavailable	Failed to complete the request because an internal service error occurs. The system is currently unavailable.
504 Gateway Timeout	A gateway timeout error occurs.

10.2 Obtaining a Project ID

Certain URLs require a project ID for calling APIs. To obtain a project ID, perform the following operations:

1. Obtain the token.
See [Token Authentication](#).
2. Obtain a project ID.

The API for obtaining the project ID is **GET https://iam.eu-west-0.myhuaweicloud.com/v3/projects**.

Add **X-Auth-Token** to the request header and set its value to the token obtained in the preceding step.

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

```
{  
  "links": {},
```



```
"projects": [  
  {  
    "is_domain": ,  
    "description": "",  
    "links": {},  
    "enabled": true,  
    "id": "", // Project ID  
    "parent_id": "",  
    "domain_id": "",  
    "name": ""  
  },  
  ...  
]
```

10.3 Task Request Result

10.3.1 Responses (Task)

- Normal response description

Parameter	Type	Description
job_id	String	Specifies the returned task ID after delivering the task. Users can query the task progress using this ID. For how to query the execution status of the task based on the task ID, see Task Status Management .

- Abnormal response description

Parameter	Type	Description
error	Object	Specifies the returned error message when an error occurs. For details, see Table 10-1 .

Table 10-1 error field structure

Parameter	Type	Description
message	String	Describes the error message when an error occurs.
code	String	Specifies the error code when an error occurs.
details	Array of objects	Specifies error details. Error details provide the error code and fault description, facilitating error handling. This field is optional.

Table 10-2 details field description

Parameter	Type	Description
message	String	Describes the error message when an error occurs. This field is optional.
code	String	Specifies the error code when an error occurs. This field is optional.

- Example response

Normal response

```
{
  "job_id": "ff80808288d41e1b018990260955686a"
}
```

Abnormal response

```
{
  "error": {"message": "", "code": "XXX",""}
}
```

Abnormal response containing error details:

```
{
  "error": {
    "message": "xxx",
    "code": "xxx",
    "details": [
      {
        "code": "xxx",
        "message": "xxx"
      }
    ]
  }
}
```

10.3.2 Returned Values

- Normal

Returned Value	Description
200	The task is successfully delivered.
202	After the task is successfully delivered, the task to be delivered shall be postponed because the system is busy.
204	The task is successfully delivered.

- Abnormal

Returned Value	Description
400 Bad Request	The server failed to process the request.

Returned Value	Description
401 Unauthorized	You need to enter the username and password to access the page requested.
403 Forbidden	You are forbidden to access the page requested.
404 Not Found	The server cannot find the page requested.
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.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then, 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 an internal service error occurs. A service exception occurred.
501 Not Implemented	Failed to complete the request because an internal service error occurs. The server does not support the requested function.
502 Bad Gateway	Failed to complete the request because an internal service error occurs. Failed to complete the request because the server receives an invalid request.
503 Service Unavailable	Failed to complete the request because an internal service error occurs. The system is currently unavailable.
504 Gateway Timeout	A gateway timeout error occurs.

10.4 Batch Task Request

10.4.1 Responses (Batch Operation)

The following responses are only for resetting the passwords for logging in to ECSs in a batch and for modifying ECS specifications in a batch. For details about the responses of other batch operations, see [Responses \(Task\)](#).

- Normal responses

Parameter	Type	Description
response	Array of objects	Specifies the response returned after a request is successfully submitted. For details, see Table 10-3 .

Table 10-3 response field description

Parameter	Type	Description
id	String	Specifies the ID of the ECS on which the operation has been successfully performed.

- Abnormal responses

Parameter	Type	Description
error	Object	Specifies the error in a batch request. For details, see Table 10-4 .
internalError	Array of objects	Specifies the error in each request among the requests submitted in a batch. For details, see Table 10-5 .

Table 10-4 error field structure

Parameter	Type	Description
message	String	Describes a batch operation error.
code	String	Specifies the code for a batch operation error.

Table 10-5 internalEroCMM.0101r field description

Parameter	Type	Description
id	String	Specifies the ID of the ECS on which a request failed.
error_message	String	Describes a single request failure.
error_code	String	Specifies the code for a single request error.

- Example response

Normal response

```
{
  "response": [
```

```
{
  "id": "616fb98f-46ca-475e-917e-2563e5a8cd19"
},
{
  "id": "516fb98f-46ca-475e-917e-2563e5a8cd12"
}
]
```

Abnormal response

```
{
  "error": {
    "code": "Ecs.xxxx",
    "message": "xxxxxxxxxxxxxxxx"
  },
  "internalError": [
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "error_code": "ECS.XXXX",
      "error_message": "xxxxxxxxxxxxxxxx"
    },
    {
      "id": "516fb98f-46ca-475e-917e-2563e5a8cd12",
      "error_code": "ECS.XXXX",
      "error_message": "xxxxxxxxxxxxxxxx"
    }
  ]
}
```

11 Out-of-Date APIs

11.1 Status Management

11.1.1 Querying Automatic Recovery of an ECS (Discarded)

Function

This API is used to query automatic recovery configured for an ECS.

URI

GET /v1/{project_id}/cloudservers/{server_id}/autorecovery

[Table 11-1](#) describes the parameters in the URI.

Table 11-1 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 11-2](#) describes the response parameters.

Table 11-2 Response parameters

Parameter	Type	Description
support_auto_recovery	String	Queries automatic recovery configured for an ECS. <ul style="list-style-type: none">• true: indicates that automatic recovery is configured for an ECS.• false: indicates that automatic recovery is not configured for an ECS.

Example Request

None

Example Response

```
GET https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/autorecovery
{
  "support_auto_recovery": "true"
}
```

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

11.1.2 Managing Automatic Recovery of an ECS (Discarded)

Function

This API is used to configure or delete automatic recovery of an ECS.

URI

PUT /v1/{project_id}/cloudservers/{server_id}/autorecovery

[Table 11-3](#) describes the parameters in the URI.

Table 11-3 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 11-4](#) describes the request parameters.

Table 11-4 Request parameters

Parameter	Mandatory	Type	Description
support_auto_recovery	Yes	String	Configures or deletes automatic recovery of an ECS. <ul style="list-style-type: none">• true: indicates configuring automatic recovery for an ECS.• false: indicates deleting automatic recovery of an ECS.

Response

None

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/autorecovery
{
  "support_auto_recovery": "true"
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

Error Codes

See [Error Codes](#).

11.2 Image Management (OpenStack Nova APIs)

11.2.1 Querying Images (Discarded)

Function

This API is used to query all images.

This API has been discarded. Use the API described in "Querying Images (Native OpenStack API)".

URI

GET /v2.1/{project_id}/images?name={name}&status={status}&changes-since={changes-since}&minRam={minRam}&minDisk={inDisk}

Table 11-5 describes the parameters in the URI.

Table 11-5 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Parameters in the following table can be used as URI parameters to filter query results. Usage: /v2/{project_id}/images? name ={name}&status={status}

Table 11-6 describes the query parameters.

Table 11-6 Query parameters

Parameter	Mandatory	Type	Description
name	No	String	Specifies the image name.
status	No	String	Specifies the image status. You cannot query images when the value is set to deleted . The value depends on the status in Glance. Table 11-7 shows the mapping relationship of image status in Nova and Glance.
changes-since	No	String	Specifies the images modified after the changes-since time point. The parameter is in ISO 8601 time format, for example, 2013-06-09T06:42:18Z.
minRam	No	Integer	Specifies the minimum memory size in MB required by the image.
minDisk	No	Integer	Specifies the minimum disk size in GB required by the image.

Table 11-7 Mapping relationship of image status in Nova and Glance

Image Status in Glance	Image Status in Nova
queued	saving
saving	saving
active	active
deleted	deleted

Request

None

Response

[Table 11-8](#) describes the response parameters.

Table 11-8 Response parameters

Parameter	Mandatory	Type	Description
images	Yes	Array of objects	Specifies the image information.
images_links	No	Array of objects	Specifies the information about the next page when you query images in pages.

Table 11-9 images parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the image ID in UUID format.
links	Yes	Array of objects	Specifies the shortcut link of the image.
name	Yes	String	Specifies the image name.

Table 11-10 images_links parameters

Parameter	Mandatory	Type	Description
href	Yes	String	Specifies the URL of the next page when you query images in pages.
rel	Yes	String	Specifies the query direction when you query images in pages.

Table 11-11 links parameters

Parameter	Mandatory	Type	Description
href	Yes	String	Specifies the link of the corresponding resource.
rel	Yes	String	The value can be: <ul style="list-style-type: none"> • self: A self link contains a version link to the resource. Use these links when the link is followed immediately. • bookmark: A bookmark link provides a permanent link to a resource, which is suitable for long term storage. • alternate: An alternate link can contain an alternate representation of the resource. For example, an OpenStack Compute image may have an alternate representation in the OpenStack image service.
type	No	String	The type attribute provides a hint as to the type of representation to expect when following the link.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/images
```

Example Response

```
{
  "images": [
    {
      "id": "ee10f19c-503c-44af-af2f-73d5e42f7a17",
      "links": [
        {
          "href": "http://xxx/v2/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
          "rel": "self"
        },
        {
          "href": "http://xxx/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
          "rel": "bookmark"
        },
        {
          "href": "http://xxx/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "name": "image1"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

11.2.2 Querying Image Details (Discarded)

Function

This API is used to query detailed information about an image list.

This API has been discarded. Use the API described in "Querying Images (Native OpenStack API)".

URI

GET /v2.1/{project_id}/images/detail?name={name}&status={status}&changes-since={changes-since}&minRam={minRam}&minDisk={inDisk}

[Table 11-12](#) describes the parameters in the URI.

Table 11-12 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Parameters in the following table can be used as URI parameters to filter query results. Usage: `/v2/{tenant_id}/images/detail? name={name}&status={status}`

Table 11-13 describes the query parameters.

Table 11-13 Query parameters

Parameter	Mandatory	Type	Description
name	No	String	Specifies the image name.
status	No	String	Specifies the image status. You cannot query images when the value is set to deleted . The value depends on the status in Glance. Table 11-14 shows the mapping relationship of image status in Nova and Glance.
changes-since	No	String	Specifies the images modified after the changes-since time point. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .
minRam	No	Integer	Specifies the minimum memory size in MB required by the image.
minDisk	No	Integer	Specifies the minimum disk size in GB required by the image.

Table 11-14 Mapping relationship of image status in Nova and Glance

Image Status in Glance	Image Status in Nova
queued	saving
saving	saving
active	active
deleted	deleted

Request

None

Response

Table 11-15 describes the response parameters.

Table 11-15 Response parameters

Parameter	Type	Description
id	String	Specifies the image ID in UUID format.
links	Array of objects	Specifies the shortcut link of the image.
name	String	Specifies the image name.
metadata	Object	Specifies the key pair of the metadata.
OS-EXT-IMG-SIZE:size	Integer	Specifies the image size. The value must be greater than zero.
minDisk	Integer	Specifies the minimum disk size in GB required by the image. The value must be greater than zero.
minRam	Integer	Specifies the minimum memory size in GB required by the image. The value must be greater than zero.
progress	Integer	Specifies the image upload progress. The value must be greater than zero.
status	String	Specifies the image status.
created	String	Specifies the image creation time. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .
updated	String	Specifies the image update time. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .

Table 11-16 links parameters

Parameter	Mandatory	Type	Description
href	Yes	String	Specifies the link of the corresponding resource.

Parameter	Mandatory	Type	Description
rel	Yes	String	The value can be: <ul style="list-style-type: none"> • self: A self link contains a version link to the resource. Use these links when the link is followed immediately. • bookmark: A bookmark link provides a permanent link to a resource, which is suitable for long term storage. • alternate: An alternate link can contain an alternate representation of the resource. For example, an OpenStack Compute image may have an alternate representation in the OpenStack image service.
type	No	String	The type attribute provides a hint as to the type of representation to expect when following the link.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/images/detail
```

Example Response

```
{
  "image": {
    "OS-EXT-IMG-SIZE:size": 20578304,
    "created": "2014-02-10T17:05:01Z",
    "id": "ee10f19c-503c-44af-af2f-73d5e42f7a17",
    "links": [
      {
        "href": "http://xxx/v2/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
        "rel": "self"
      },
      {
        "href": "http://xxx/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
        "rel": "bookmark"
      },
      {
        "href": "http://xxx/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
        "rel": "alternate",
        "type": "application/vnd.openstack.image"
      }
    ]
  }
}
```

```
],
  "metadata": {
    "clean_attempts": "3",
    "image_location": "snapshot",
    "image_state": "available",
    "image_type": "snapshot",
    "instance_type_ephemeral_gb": "0",
    "instance_type_flavorid": "6",
    "instance_type_id": "7",
    "instance_type_memory_mb": "256",
    "instance_type_name": "wj.ssd",
    "instance_type_root_gb": "2",
    "instance_type_rxtx_factor": "1.0",
    "instance_type_swap": "0",
    "instance_type_vcpus": "1",
    "instance_uuid": "b600b5b1-ed8c-4814-aefa-8b903c894c20",
    "os_type": "None",
    "owner_id": "d9ebe43510414ef590a4aa158605329e",
    "user_id": "74fe4ff0674b434b8a274077d8106c5b"
  },
  "minDisk": 2,
  "minRam": 0,
  "name": "image1",
  "progress": 100,
  "server": {
    "id": "b600b5b1-ed8c-4814-aefa-8b903c894c20",
    "links": [
      {
        "href": "http://xxx/v2/d9ebe43510414ef590a4aa158605329e/servers/b600b5b1-ed8c-4814-aefa-8b903c894c20",
        "rel": "self"
      },
      {
        "href": "http://xxx/d9ebe43510414ef590a4aa158605329e/servers/b600b5b1-ed8c-4814-aefa-8b903c894c20",
        "rel": "bookmark"
      }
    ]
  },
  "status": "ACTIVE",
  "updated": "2014-02-10T17:05:07Z"
}
```

Returned Values

See [Returned Values for General Requests](#).

11.2.3 Querying Details About a Specified Image (Discarded)

Function

This API is used to query the details about the specified image.

This API has been discarded. Use the API described in "Querying Images (Native OpenStack API)".

URI

GET /v2.1/{project_id}/images/{image_id}

[Table 11-17](#) describes the parameters in the URI.

Table 11-17 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
image_id	Yes	Specifies the image ID.

Request

None

Response

[Table 11-18](#) describes the response parameters.

Table 11-18 Response parameters

Parameter	Type	Description
id	String	Specifies the image ID in UUID format.
links	Array of objects	Specifies the shortcut link of the image.
name	String	Specifies the image name.
metadata	Object	Specifies the key pair of the metadata.
OS-EXT-IMG-SIZE:size	Integer	Specifies the image size. The value must be greater than zero.
minDisk	Integer	Specifies the minimum disk size in GB required by the image. The value must be greater than zero.
minRam	Integer	Specifies the minimum memory size in GB required by the image. The value must be greater than zero.
progress	Integer	Specifies the image upload progress. The value must be greater than zero.
status	String	Specifies the image status.
created	String	Specifies the image creation time. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .

Parameter	Type	Description
updated	String	Specifies the image update time. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .

Table 11-19 links parameters

Parameter	Mandator y	Type	Description
href	Yes	String	Specifies the link of the corresponding resource.
rel	Yes	String	The value can be: <ul style="list-style-type: none"> • self: A self link contains a version link to the resource. Use these links when the link is followed immediately. • bookmark: A bookmark link provides a permanent link to a resource, which is suitable for long term storage. • alternate: An alternate link can contain an alternate representation of the resource. For example, an OpenStack Compute image may have an alternate representation in the OpenStack image service.
type	No	String	The type attribute provides a hint as to the type of representation to expect when following the link.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/images/17a1890b-0fa4-485e-8505-14e294017988
```

Example Response

```
{
  "image": {
    "status": "ACTIVE",
    "updated": "2015-12-27T02:52:25Z",
    "name": "cirror",
```

```
"links": [
  {
    "href": "https://compute.localdomain.com:8001/v2/719e9483f42d4784a089862ac4c3e8d0/
images/17a1890b-0fa4-485e-8505-14e294017988",
    "rel": "self"
  },
  {
    "href": "https://compute.localdomain.com:8001/719e9483f42d4784a089862ac4c3e8d0/images/
17a1890b-0fa4-485e-8505-14e294017988",
    "rel": "bookmark"
  },
  {
    "href": "https://https://
image.az2.dc1.domainname.com:443/719e9483f42d4784a089862ac4c3e8d0/images/
17a1890b-0fa4-485e-8505-14e294017988",
    "type": "application/vnd.openstack.image",
    "rel": "alternate"
  }
],
"created": "2015-12-27T02:52:24Z",
"minDisk": 0,
"progress": 100,
"minRam": 0,
"metadata": {
  "_os_version": "CentOS 4.4 32bit",
  "file_format": "img",
  "file_name": "**.img",
  "describe": "",
  "_os_type": "Linux",
  "virtual_env_type": "KVM",
  "hw_disk_bus": "scsi"
},
"id": "17a1890b-0fa4-485e-8505-14e294017988",
"OS-EXT-IMG-SIZE:size": 13167616
}
```

Returned Values

See [Returned Values for General Requests](#).

11.2.4 Querying the Metadata of a Specified Image (Discarded)

Function

This API is used to query the metadata of the specified image.

This API has been discarded. Use the API described in "Querying Image Metadata (Native OpenStack API)".

URI

GET /v2.1/{project_id}/images/{image_id}/metadata

[Table 11-20](#) describes the parameters in the URI.

Table 11-20 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
image_id	Yes	Specifies the image ID.

Request

None

Response

[Table 11-21](#) describes the response parameters.

Table 11-21 Response parameters

Parameter	Type	Description
User customization	String	Specifies the key pair of the metadata.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/images/  
17a1890b-0fa4-485e-8505-14e294017988/metadata
```

Example Response

```
{  
  "metadata": {  
    "__os_version": "Suse Linux Enterprise 12.2 64bit",  
    "__image_source_type": "uds",  
    "__imagetype": "gold",  
    "__os_bit": "64",  
    "__os_type": "Suse",  
    "__isregistered": "true",  
    "__image_location": "192.168.80.11:5080:pcsimbeta:suse12.2-addx710-05-11",  
    "virtual_env_type": "Ironic",  
    "__platform": "Suse",  
    "__support_o3s": "true"  
  }  
}
```

Returned Values

See [Returned Values for General Requests](#).

11.2.5 Deleting an Image (Discarded)

Function

This API is used to delete a specified image. The image cannot be restored after being deleted.

This API has been discarded. Use the API described in "Deleting an Image (Native OpenStack API)".

URI

DELETE /v2.1/{project_id}/images/{image_id}

[Table 11-22](#) describes the parameters in the URI.

Table 11-22 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
image_id	Yes	Specifies the image ID.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/images/6cad483b-e281-4985-a345-7afef1f3c5b7
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.3 Security Group Management (OpenStack Nova APIs)

11.3.1 Querying Security Groups (Discarded)

Function

This API is used to query security groups.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Querying Security Groups" in *Virtual Private Network API Reference*.

URI

GET /v2.1/{project_id}/os-security-groups

[Table 11-23](#) describes the parameters in the URI.

Table 11-23 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

 **NOTE**

Pagination query is not supported.

Request

N/A

Response

[Table 11-24](#) describes the response parameters.

Table 11-24 Response parameters

Parameter	Type	Description
security_groups	Array of objects	Specifies security groups. For details, see Table 11-25 .

Table 11-25 security_group objects

Parameter	Type	Description
description	String	Specifies information about a security group. It must contain 0 to 255 characters.
id	String	Specifies the security group ID in UUID format.
name	String	Specifies the security group name. It must contain 0 to 255 characters.
rules	Array of objects	Specifies security group rules. For details, see Table 11-26 .
tenant_id	String	Specifies the tenant or project ID.

Table 11-26 security_group_rule objects

Parameter	Type	Description
parent_group_id	String	Specifies the associated security group ID in UUID format.
ip_protocol	String	Specifies the protocol type or the IP protocol number. The value can be icmp , tcp , udp , or the IP protocol number.
from_port	Integer	<p>Specifies the start port number. The value ranges from 1 to 65,535 and cannot be greater than to_port.</p> <p>When ip_protocol is icmp, this parameter indicates the ICMP type field with a length from 0 to 255 characters.</p> <p>NOTE</p> <p>The ICMP message type is determined by the type field and code field in the packet. For details, see Appendix > ICMP-Port Range Relationship Table in <i>Virtual Private Cloud API Reference</i>.</p> <p>port_range_min indicates the ICMP type field, and port_range_max indicates the ICMP code field.</p>

Parameter	Type	Description
to_port	Integer	<p>Specifies the stop port number. The value ranges from 1 to 65,535 and cannot be less than from_port.</p> <p>When ip_protocol is icmp, this parameter indicates the ICMP code field with a length from 0 to 255 characters.</p> <p>NOTE The ICMP message type is determined by the type field and code field in the packet. For details, see Appendix > ICMP-Port Range Relationship Table in <i>Virtual Private Cloud API Reference</i>. port_range_min indicates the ICMP type, and port_range_max indicates the ICMP code.</p>
ip_range	Object	<p>Specifies the peer IP segment in CIDR format. For details, see Table 11-27.</p> <p>Specify either ip_range or group.</p>
group	Object	<p>Specifies the name of the peer security group and the ID of the tenant in the peer security group. For details, see Table 11-28.</p> <p>Specify either ip_range or group.</p>
id	String	<p>Specifies the security group rule ID in UUID format.</p>

Table 11-27 ip_range objects

Parameter	Type	Description
cidr	String	<p>Specifies the peer IP segment in CIDR format.</p>

Table 11-28 group objects

Parameter	Type	Description
tenant_id	String	<p>Specifies the ID of the tenant of the peer security group.</p>
name	String	<p>Specifies the name of the peer security group.</p>

Example Request

```
GET https://{endpoint}/v2.1/bb1118612ba64af3a6ea63a1bdcaa5ae/os-security-groups
```

Example Response

```
{
  "security_groups": [
    {
      "rules": [
        {
          "from_port": null,
          "group": {
            "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
            "name": "default"
          },
          "ip_protocol": null,
          "to_port": null,
          "parent_group_id": "bc4ac1d1-dc77-4b7d-a97d-af86eb0dc450",
          "ip_range": {},
          "id": "bb3cc988-e06a-49f6-b668-600e8bf193ee"
        },
        {
          "from_port": null,
          "group": {
            "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
            "name": "default"
          },
          "ip_protocol": null,
          "to_port": null,
          "parent_group_id": "bc4ac1d1-dc77-4b7d-a97d-af86eb0dc450",
          "ip_range": {},
          "id": "f9371051-d7e1-4be4-8748-77b1e0913730"
        }
      ],
      "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
      "description": "default",
      "id": "bc4ac1d1-dc77-4b7d-a97d-af86eb0dc450",
      "name": "default"
    },
    {
      "rules": [
        {
          "from_port": 200,
          "group": {},
          "ip_protocol": "tcp",
          "to_port": 400,
          "parent_group_id": "b3e4b615-a40f-4e1c-92af-2e0d382141d5",
          "ip_range": {
            "cidr": "0.0.0.0/0"
          },
          "id": "3330120d-bbd1-4a73-bda9-0196a84d5670"
        },
        {
          "from_port": 201,
          "group": {},
          "ip_protocol": "tcp",
          "to_port": 400,
          "parent_group_id": "b3e4b615-a40f-4e1c-92af-2e0d382141d5",
          "ip_range": {
            "cidr": "0.0.0.0/0"
          },
          "id": "b550c9a6-970a-462d-984e-265e88020818"
        }
      ],
      "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
```

```
"description": "desc-sg",  
"id": "b3e4b615-a40f-4e1c-92af-2e0d382141d5",  
"name": "test-sg"  
}  
]  
}
```

Returned Values

See [Returned Values for General Requests](#).

11.3.2 Creating a Security Group (Discarded)

Function

This API is used to create a security group.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Creating a Security Group" in *Virtual Private Network API Reference*.

URI

POST /v2.1/{project_id}/os-security-groups

[Table 11-29](#) describes the parameters in the URI.

Table 11-29 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 11-30](#) describes the request parameters.

Table 11-30 Request parameters

Parameter	Mandatory	Type	Description
security_group	Yes	Object	Specifies the security group, which is configured in the message body. For details, see Table 11-31 .

Table 11-31 Objects of request parameter **security_group**

Parameter	Mandatory	Type	Description
name	No	String	Specifies the security group name. It must contain 0 to 255 characters.
description	No	String	Specifies information about a security group. It must contain 0 to 255 characters.

Response

Table 11-32 describes the response parameters.

Table 11-32 Response parameters

Parameter	Type	Description
security_group	Object	Specifies the security group. For details, see Table 11-33 .

Table 11-33 Objects of response parameter **security_group**

Parameter	Type	Description
description	String	Provides supplementary information about the security group.
id	String	Specifies the security group ID in UUID format.
name	String	Specifies the security group name.
rules	Array of objects	Specifies the rules of the security group. The list is empty.
tenant_id	String	Specifies the tenant or project ID.

Example Request

```
POST https://{endpoint}/v2.1/bb1118612ba64af3a6ea63a1bdcaa5ae/os-security-groups
{
  "security_group": {
    "name": "test",
    "description": "description"
  }
}
```

```
}  
}
```

Example Response

```
{  
  "security_group": {  
    "rules": [],  
    "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",  
    "description": "desc-sg",  
    "id": "81f1d23b-b1e2-42cd-bdee-359b4a065a42",  
    "name": "test-sg"  
  }  
}
```

Returned Values

See [Returned Values for General Requests](#).

11.3.3 Querying Details About a Security Group (Discarded)

Function

This API is used to query details about a security group.

This API can only query the inbound security group rules. To query the outbound security group rules, see "Querying a Security Group" in "Security Group (Native OpenStack API)" in the *Virtual Private Cloud API Reference*.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Querying a Security Group" in *Virtual Private Network API Reference*.

URI

GET /v2.1/{project_id}/os-security-groups/{security_group_id}

[Table 11-34](#) describes the parameters in the URI.

Table 11-34 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
security_group_id	Yes	Specifies the security group ID, which is specified in the URI.

Request

None

Response

[Table 11-35](#) describes the response parameters.

Table 11-35 Response parameters

Parameter	Type	Description
security_group	Object	Specifies the security group. For details, see Table 11-36 .

Table 11-36 security_group objects

Parameter	Type	Description
description	String	Specifies information about a security group. It must contain 0 to 255 characters.
id	String	Specifies the security group ID in UUID format.
name	String	Specifies the security group name. It must contain 0 to 255 characters.
rules	Array of objects	Specifies security group rules. For details, see Table 11-37 .
tenant_id	String	Specifies the tenant or project ID.

Table 11-37 security_group_rule objects

Parameter	Type	Description
parent_group_id	String	Specifies the associated security group ID in UUID format.
ip_protocol	String	Specifies the protocol type or the IP protocol number. The value can be icmp , tcp , udp , or the IP protocol number.

Parameter	Type	Description
from_port	Integer	<p>Specifies the start port number. The value ranges from 1 to 65,535 and cannot be greater than to_port.</p> <p>When ip_protocol is icmp, this parameter indicates the ICMP type field with a length from 0 to 255 characters.</p> <p>NOTE The ICMP message type is determined by the type field and code field in the packet. For details, see Appendix > ICMP-Port Range Relationship Table in <i>Virtual Private Cloud API Reference</i>. port_range_min indicates the ICMP type, and port_range_max indicates the ICMP code.</p>
to_port	Integer	<p>Specifies the stop port number. The value ranges from 1 to 65,535 and cannot be less than from_port.</p> <p>When ip_protocol is icmp, this parameter indicates the ICMP code field with a length from 0 to 255 characters.</p> <p>NOTE The ICMP message type is determined by the type field and code field in the packet. For details, see Appendix > ICMP-Port Range Relationship Table in <i>Virtual Private Cloud API Reference</i>. port_range_min indicates the ICMP type, and port_range_max indicates the ICMP code.</p>
ip_range	Object	<p>Specifies the peer IP segment in CIDR format. For details, see Table 11-38.</p> <p>Specify either ip_range or group.</p>
group	Object	<p>Specifies the name of the peer security group and the ID of the tenant in the peer security group. For details, see Table 11-39.</p> <p>Specify either ip_range or group.</p>

Parameter	Type	Description
id	String	Specifies the security group rule ID.

Table 11-38 ip_range objects

Parameter	Type	Description
cidr	String	Specifies the peer IP segment in CIDR format.

Table 11-39 group objects

Parameter	Type	Description
tenant_id	String	Specifies the ID of the tenant of the peer security group.
name	String	Specifies the name of the peer security group.

Example Request

```
GET https://{endpoint}/v2.1/bb1118612ba64af3a6ea63a1bdcaa5ae/os-security-groups/81f1d23b-b1e2-42cd-bdee-359b4a065a42
```

Example Response

```
{
  "security_group": {
    "rules": [],
    "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
    "id": "81f1d23b-b1e2-42cd-bdee-359b4a065a42",
    "name": "test-sg",
    "description": "desc-sg"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.3.4 Updating a Security Group (Discarded)

Function

This API is used to update a security group.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Updating a Security Group" in *Virtual Private Network API Reference*.

URI

PUT /v2.1/{project_id}/os-security-groups/{security_group_id}

[Table 11-40](#) describes the parameters in the URI.

Table 11-40 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
security_group_id	Yes	Specifies the security group ID, which is specified in the URI.

Request

[Table 11-41](#) describes the request parameters.

Table 11-41 Request parameters

Parameter	Mandatory	Type	Description
security_group	Yes	Object	Specifies the security group in the message body. For details, see Table 11-42 .

Table 11-42 Objects of request parameter **security_group**

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the security group name. The value cannot exceed 255 characters.
description	Yes	String	Specifies information about a security group. The value cannot exceed 255 characters.

Response

[Table 11-43](#) describes the response parameters.

Table 11-43 Response parameters

Parameter	Mandatory	Type	Description
security_group	Yes	Object	Specifies the security group. For details, see Table 11-44 .

Table 11-44 Objects of response parameter `security_group`

Parameter	Mandatory	Type	Description
description	Yes	String	Specifies information about a security group. The value cannot exceed 255 characters.
id	Yes	String	Specifies the security group ID in UUID format.
name	Yes	String	Specifies the security group name. The value cannot exceed 255 characters.
rules	Yes	Array of objects	Specifies the security group rule list. For details, see Table 11-45 .
tenant_id	Yes	String	Specifies the tenant or project ID. The value cannot exceed 255 characters.

Table 11-45 `security_group_rule` objects

Parameter	Mandatory	Type	Description
parent_group_id	Yes	String	Specifies the associated security group ID in UUID format.
ip_protocol	Yes	String	Specifies the protocol type or the IP protocol number. The value can be icmp , tcp , udp , or the IP protocol number.

Parameter	Mandatory	Type	Description
from_port	Yes	Integer	Specifies the start port. The value ranges from 1 to 65,535 and cannot be greater than to_port . When ip_protocol is icmp , this parameter specifies a port type with a length from 0 to 255 characters.
to_port	Yes	Integer	Specifies the end port. The value ranges from 1 to 65,535 and cannot be less than from_port . When ip_protocol is icmp , it specifies the code. The value ranges from 0 to 255. If both from_port and to_port are -1 , any ICMP packet can be transmitted.
ip_range	Yes	Object	Specifies the peer IP segment in CIDR format. For details, see Table 11-46 . The value of ip_range or group must be empty.
group	Yes	Object	Specifies the name of the peer security group and the ID of the tenant in the peer security group. For details, see Table 11-47 . The value of ip_range or group must be empty.
id	Yes	String	Specifies the security group rule ID in UUID format.

Table 11-46 ip_range objects

Parameter	Mandatory	Type	Description
cidr	Yes	String	Specifies the peer IP segment in CIDR format. The value cannot exceed 255 characters.

Table 11-47 group objects

Parameter	Mandatory	Type	Description
tenant_id	Yes	String	Specifies the ID of the tenant of the peer security group.
name	Yes	String	Specifies the name of the peer security group.

Example Request

```
PUT https://{endpoint}/v2.1/bb1118612ba64af3a6ea63a1bdcaa5ae/os-security-groups/  
3d02312d-0764-49c9-8244-2368ddce0045  
{  
  "security_group": {  
    "name": "test",  
    "description": "description"  
  }  
}
```

Example Response

```
{  
  "security_group": {  
    "rules": [  
      {  
        "from_port": null,  
        "group": {  
          "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",  
          "name": "test"  
        },  
        "ip_protocol": null,  
        "to_port": null,  
        "parent_group_id": "3d02312d-0764-49c9-8244-2368ddce0045",  
        "ip_range": {},  
        "id": "00dec0b6-8e96-4906-aadf-46cfe54cf5ef"  
      }  
    ],  
    "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",  
    "id": "3d02312d-0764-49c9-8244-2368ddce0045",  
    "name": "test",  
    "description": "description"  
  }  
}
```

Returned Values

See [Returned Values for General Requests](#).

11.3.5 Deleting a Security Group (Discarded)

Function

This API is used to delete a security group.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Deleting a Security Group" in *Virtual Private Network API Reference*.

URI

DELETE /v2.1/{project_id}/os-security-groups/{security_group_id}

[Table 11-48](#) describes the parameters in the URI.

Table 11-48 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
security_group_id	Yes	Specifies the security group ID, which is specified in the URI.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/bb1118612ba64af3a6ea63a1bdcaa5ae/os-security-groups/81f1d23b-b1e2-42cd-bdee-359b4a065a42
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.3.6 Creating a Security Group Rule (Discarded)

Function

This API is used to create a security group rule.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Creating a Security Group Rule" in *Virtual Private Network API Reference*.

URI

POST /v2.1/{project_id}/os-security-group-rules

[Table 11-49](#) describes the parameters in the URI.

Table 11-49 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 11-50](#) describes the request parameters.

Table 11-50 Request parameters

Parameter	Mandatory	Type	Description
security_group_rule	Yes	Object	Specifies the security group rule, which is configured in the message body. For details, see Table 11-51 .

Table 11-51 Objects of request parameter **security_group_rule**

Parameter	Mandatory	Type	Description
parent_group_id	Yes	String	Specifies the associated security group ID in UUID format.
ip_protocol	Yes	String	Specifies the IP protocol, which can be icmp , tcp , or udp .
from_port	Yes	Integer	Specifies the start port. The value ranges from 1 to 65,535 and is no greater than the value of to_port . If the value of ip_protocol is icmp , this parameter specifies the ICMP type. The value ranges from 0 to 255 .

Parameter	Mandatory	Type	Description
to_port	Yes	Integer	Specifies the end port. The value ranges from 1 to 65,535 and cannot be less than from_port . If ip_protocol is icmp , this parameter specifies the ICMP code. The value ranges from 0 to 255. If both from_port and to_port are -1 , any ICMP packet can be transmitted.
cidr	No	String	Specifies the IP address range. The address is in CIDR format, such as 192.168.0.0/24.
group_id	No	String	Specifies the source security group ID. If both group_id and cidr are set, group_id is used.

Response

[Table 11-52](#) describes the response parameters.

Table 11-52 Response parameters

Parameter	Mandatory	Type	Description
security_group_rule	Yes	Object	Specifies the security group rule, which is configured in the message body. For details, see Table 11-53 .

Table 11-53 Objects of response parameter **security_group_rule**

Parameter	Mandatory	Type	Description
parent_group_id	Yes	String	Specifies the associated security group ID in UUID format.
ip_protocol	Yes	String	Specifies the IP protocol, which can be icmp , tcp , or udp .

Parameter	Mandatory	Type	Description
from_port	Yes	Integer	Specifies the start port number. The value ranges from 1 to 65,535 and cannot be greater than to_port . When the protocol type is set to ICMP, from_port is the ICMP type and ranges from 0 to 255.
to_port	Yes	Integer	Specifies the end port number. The value ranges from 1 to 65,535 . <ul style="list-style-type: none">When the protocol type is set to ICMP, to_port is the ICMP code and ranges from 0 to 255.If both from_port and to_port are -1, it indicates that any ICMP packet can be transmitted.
ip_range	Yes	Object	Specifies the IP address range, including the CIDR information, such as " ip_range ": {"cidr": " 0.0.0.0/0 "}. For details, see the ip_range object.
group	Yes	Object	Nothing is returned.
id	Yes	String	Specifies the security group rule ID in UUID format.

Table 11-54 ip_range objects

Parameter	Mandatory	Type	Description
cidr	Yes	String	Specifies the IP address range. The address is in CIDR format, such as 192.168.0.0/24.

Example Request

```
POST https://{endpoint}/v2.1/{project_id}/os-security-group-rules
{
  "security_group_rule": {
    "from_port": "443",
    "ip_protocol": "tcp",
```

```
    "to_port": "443",
    "cidr": "0.0.0.0/0",
    "parent_group_id": "48700ff3-30b8-4e63-845f-a79c9633e9fb"
  }
}
```

Example Response

```
{
  "security_group_rule": {
    "id": "F4966B29-D21D-B211-B6B4-0018E1C5D866",
    "ip_range": {
      "cidr": "0.0.0.0/0"
    },
    "parent_group_id": "48700ff3-30b8-4e63-845f-a79c9633e9fb",
    "to_port": 443,
    "ip_protocol": "tcp",
    "group": {
      },
    },
    "from_port": 443
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.3.7 Deleting a Security Group Rule (Discarded)

Function

This API is used to delete a security group rule.

This API has been discarded. Use the API described in section "Security Group (OpenStack Neutron APIs) > Deleting a Security Group Rule" in *Virtual Private Network API Reference*.

URI

DELETE /v2.1/{project_id}/os-security-group-rules/{security_group_rule_id}

[Table 11-55](#) describes the parameters in the URI.

Table 11-55 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
security_group_rule_id	Yes	Specifies the security group rule ID, which is specified in the URI.

Request

None

Response

None

Example Request

Example request

```
DELETE https://{endpoint}/v2.1/3d72597871904daeb6887f75f848b531/os-security-group-rules/012fa2c6-  
bf4a-4b0b-b893-70d0caee81c7
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.4 Disk Management (OpenStack Nova APIs)

11.4.1 Querying Brief Information About Disks (Discarded)

Function

This API is used to query brief information about disks.

This API has been discarded. Use the EVS API "Querying EVS Disks (OpenStack Cinder API v2)".

URI

GET /v2.1/{project_id}/os-volumes

[Table 11-56](#) describes the parameters in the URI.

Table 11-56 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

[Table 11-57](#) describes the response parameters.

Table 11-57 Response parameters

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.
displayName	String	Specifies the disk name.
status	String	Specifies the disk status.
attachments	Array of objects	Specifies the attachment information about a disk.
availabilityZone	String	Specifies the AZ to which the disk belongs.
createdAt	String	Specifies the time when the disk was created.
displayDescription	String	Specifies the disk description.
volumeType	String	Specifies the disk type.
snapshotId	String	Specifies the snapshot ID.
metadata	Object	Specifies the disk metadata.
size	Integer	Specifies the disk size.

Table 11-58 attachments field description

Parameter	Type	Description
device	String	Specifies the directory to which the disk is mounted.
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ECS ID.
volumeld	String	Specifies the ID of the attached disk.

Example Request

```
GET https://{endpoint}/v2.1/b84c367e4d1047fc9b54f28b400ddbc2/os-volumes
```

Example Response

```
{
  "volumes": [
    {
      "status": "available",
      "attachments": [],
      "availabilityZone": "nova",
      "createdAt": "2016-05-20T07:57:56.299000",
      "displayDescription": null,
      "volumeType": null,
      "displayName": "test",
      "snapshotId": null,
      "metadata": {},
      "id": "70b14513-faad-4646-b7ab-a065cef282b4",
      "size": 1
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

11.4.2 Querying Detailed Information About Disks (Discarded)

Function

This API is used to query detailed information about disks.

This API has been discarded. Use the EVS API "Querying Details About All Disks (OpenStack Cinder API v2)".

URI

```
GET /v2.1/{project_id}/os-volumes/detail
```

[Table 11-59](#) describes the parameters in the URI.

Table 11-59 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

N/A

Response

Table 11-60 describes the response parameters.

Table 11-60 Response parameters

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.
displayName	String	Specifies the disk name.
status	String	Specifies the disk status.
attachments	Array of objects	Specifies the attachment information about a disk.
availabilityZone	String	Specifies the AZ to which the disk belongs.
createdAt	String	Specifies the time when the disk was created.
displayDescription	String	Specifies the disk description.
volumeType	String	Specifies the disk type.
snapshotId	String	Specifies the snapshot ID.
metadata	Object	Specifies the disk metadata.
size	Integer	Specifies the disk size.

Table 11-61 attachments field description

Parameter	Type	Description
device	String	Specifies the directory to which the disk is mounted.
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ECS ID.
volumeId	String	Specifies the ID of the attached disk.

Example Request

```
GET https://{endpoint}/v2.1/b84c367e4d1047fc9b54f28b400ddbc2/os-volumes/detail
```

Example Response

```
{  
  "volumes": [  

```

```
{
  "status": "available",
  "attachments": [],
  "availabilityZone": "nova",
  "createdAt": "2016-05-20T07:57:56.299000",
  "displayDescription": null,
  "volumeType": null,
  "displayName": "test",
  "snapshotId": null,
  "metadata": {},
  "id": "70b14513-faad-4646-b7ab-a065cef282b4",
  "size": 1
}
```

Returned Values

See [Returned Values for General Requests](#).

11.4.3 Querying Information About a Disk (Discarded)

Function

This API is used to query information about a specified disk.

This API has been discarded. Use the EVS API "Querying Details About a Disk (OpenStack Cinder API v2)".

URI

GET /v2.1/{project_id}/os-volumes/{volume_id}

[Table 11-62](#) describes the parameters in the URI.

Table 11-62 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
volume_id	Yes	Specifies the disk ID.

Request

None

Response

[Table 11-63](#) describes the response parameters.

Table 11-63 Response parameters

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.
displayName	String	Specifies the disk name.
status	String	Specifies the disk status.
attachments	Array of objects	Specifies the attachment information about a disk.
availabilityZone	String	Specifies the AZ to which the disk belongs.
createdAt	String	Specifies the time when the disk was created.
displayDescription	String	Specifies the disk description.
volumeType	String	Specifies the disk type.
snapshotId	String	Specifies the snapshot ID.
metadata	Object	Specifies the disk metadata.
size	Integer	Specifies the disk size.

Table 11-64 attachments field description

Parameter	Type	Description
device	String	Specifies the directory to which the disk is mounted.
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ECS ID.
volumeId	String	Specifies the ID of the attached disk.

Example Request

```
GET https://{endpoint}/v2.1/b84c367e4d1047fc9b54f28b400ddbc2/os-volumes/70b14513-faad-4646-b7ab-a065cef282b4
```

Example Response

```
{  
  "volume":
```

```
{
  "status": "available",
  "attachments": [],
  "availabilityZone": "nova",
  "createdAt": "2016-05-20T07:57:56.299000",
  "displayDescription": null,
  "volumeType": null,
  "displayName": "test",
  "snapshotId": null,
  "metadata": {},
  "id": "70b14513-faad-4646-b7ab-a065cef282b4",
  "size": 1
}
```

Returned Values

See [Returned Values for General Requests](#).

11.4.4 Creating a Disk (Discarded)

Function

This API is used to create a disk.

This API has been discarded. Use the EVS API "Creating EVS Disks (OpenStack Cinder API v2)".

URI

POST /v2.1/{project_id}/os-volumes

[Table 11-65](#) describes the parameters in the URI.

Table 11-65 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 11-66](#) describes the request parameters.

Table 11-66 Request parameters

Parameter	Mandatory	Type	Description
availability_zone	No	String	Specifies the AZ where the volume is created. If the specified AZ does not exist, creating the volume failed, and the volume is in error state. An AZ must be specified during the volume creation.
display_description	No	String	Specifies the volume description.
snapshot_id	No	String	Specifies the snapshot ID. If this parameter is specified, the volume is to be created from a snapshot.
size	Yes (If the volume is created from a snapshot, this parameter is optional.)	Integer	Specifies the volume size. Unit: GB
display_name	No	String	Specifies the volume name.
volume_type	No	String	Specifies the volume type.
metadata	No	Object	Specifies the volume metadata.

Response

[Table 11-67](#) describes the response parameters.

Table 11-67 Response parameters

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.
displayName	String	Specifies the volume name.
status	String	Specifies the volume status.

Parameter	Type	Description
attachments	Array of objects	Specifies the volume attachment information.
availabilityZone	String	Specifies the AZ to which the volume belongs.
createdAt	String	Specifies the time when the volume was created.
displayDescription	String	Specifies the volume description.
volumeType	String	Specifies the volume type.
snapshotId	String	Specifies the snapshot ID.
metadata	Object	Specifies the volume metadata.
size	Integer	Specifies the size of the volume.

Example Request

```
POST https://{endpoint}/v2.1/b84c367e4d1047fc9b54f28b400ddbc2/os-volumes
{
  "volume": {
    "availability_zone": "az1-dc1",
    "display_description": "test1",
    "snapshot_id": null,
    "size": 1,
    "display_name": "test",
    "volume_type": "SSD",
    "metadata": {
      "testkey": "testvalue"
    }
  }
}
```

Example Response

```
{
  "volume": {
    "displayDescription": "test1",
    "volumeType": "SATA",
    "createdAt": "2018-05-18T01:17:03.871808",
    "metadata": {
      "testkey": "testvalue",
      "resourceSpecCode": "SATA"
    },
    "attachments": [
      {}
    ],
    "snapshotId": null,
    "size": 1,
    "displayName": "test",
    "id": "b4fb891c-c665-4478-92b0-8a7fa65a57cd",
    "availabilityZone": "az1.dc1",
    "status": "creating"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.4.5 Deleting a Disk (Discarded)

Function

This API is used to delete a specified disk.

This API has been discarded. Use the EVS API "Deleting an EVS Disk (OpenStack Cinder API v2)".

Constraints

- If the volume has a snapshot not deleted, the volume cannot be deleted.
- A volume that is being attached to an ECS cannot be deleted.
- A volume that is being migrated cannot be deleted.
- Only a volume in the available, error, error_restoring, or error_extending state can be deleted.

URI

DELETE /v2.1/{project_id}/os-volumes/{volume_id}

[Table 11-68](#) describes the parameters in the URI.

Table 11-68 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
volume_id	Yes	Specifies the volume ID.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/b84c367e4d1047fc9b54f28b400ddbc2/os-volumes/0cf90bab-c513-46df-8559-45ba6de80e3f
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.5 Floating IP Address Management (OpenStack Nova APIs)

11.5.1 Binding a Floating IP Address (Discarded)

Function

This API is used to bind a floating IP address for an ECS.

This API has been discarded. Since microversion 2.44, the system will return error 404 when you call this API. You are advised to use the VPC API "Updating a Floating IP Address".

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 11-69](#) describes the parameters in the URI.

Table 11-69 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 11-70](#) describes the request parameters.

Table 11-70 Request parameter

Parameter	Mandatory	Type	Description
addFloatingIp	Yes	Object	Specifies the floating IP address to be bound to an ECS.

Table 11-71 addFloatingIp parameter information

Parameter	Mandatory	Type	Description
address	Yes	String	Specifies the floating IP address.
fixed_address	No	String	Specifies the fixed IP address with which the floating IP address associates.

Response

None

Example Request

```
POST https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/47e9be4e-a7b9-471f-92d9-ffc83814e07a/action
{
  "addFloatingIp": {
    "address": "10.144.2.1",
    "fixed_address": "192.168.1.3"
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.5.2 Unbinding a Floating IP Address (Discarded)

Function

This API is used to unbind a floating IP address from an ECS.

This API has been discarded. Since microversion 2.44, the system will return error 404 when you call this API. You are advised to use the VPC API "Updating a Floating IP Address".

URI

POST /v2.1/{project_id}/servers/{server_id}/action

[Table 11-72](#) describes the parameters in the URI.

Table 11-72 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 11-73](#) describes the request parameters.

Table 11-73 Request parameter

Parameter	Mandatory	Type	Description
removeFloatingIp	Yes	Object	Unbinds a floating IP address from an ECS.

Table 11-74 removeFloatingIp parameter information

Parameter	Mandatory	Type	Description
address	Yes	String	Specifies the floating IP address.

Response

None

Example Request

```
POST https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/47e9be4e-a7b9-471f-92d9-ffc83814e07a/action
{
  "removeFloatingIp": {
    "address": "10.144.2.1"
  }
}
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.5.3 Assigning a Floating IP Address (Discarded)

Function

This API is used to assign a floating IP address.

This API has been discarded. Use the API described in "Assigning a Floating IP Address".

Constraints

You need to obtain a network resource pool that provides floating IP addresses. To do so, run **GET /v2.0/networks?router:external=True** or **neutron net-external-list**.

URI

POST /v2.1/{project_id}/os-floating-ips

[Table 11-75](#) describes the parameters in the URI.

Table 11-75 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 11-76](#) describes the request parameters.

Table 11-76 Request parameters

Parameter	Type	Mandatory	Description
tenant_id	String	Yes	Specifies the tenant ID specified in the URI. The value is in UUID format.
pool	String	No	Specifies the network resource pool that provides floating IP addresses. If it is not specified, the default resource pool is used.

Response

[Table 11-77](#) describes the response parameters.

Table 11-77 Response parameters

Parameter	Mandatory	Type	Description
floating_ip	Yes	Object	Specifies the floating IP address. For details, see Table 11-78 .

Table 11-78 floating_ip objects

Parameter	Mandatory	Type	Description
fixed_ip	Yes	String	Specifies a private IP address.
id	Yes	String	Specifies the floating IP address ID in UUID format.
instance_id	Yes	String	Specifies the ID of a bound ECS in UUID format.
ip	Yes	String	Specifies the floating IP address.
pool	Yes	String	Specifies the name of a network resource pool that provides floating IP addresses.

Example Request

```
POST https://{endpoint}/v2.1/e73621affb8f44e1bc01898747ca09d4/os-floating-ips
{
  "pool": "external"
}
```

Example Response

```
{
  "floating_ip": {
    "id": "7aa2aa63-3097-4cfe-a2e4-596c301d3b1b",
    "pool": "external",
    "ip": "10.154.53.184",
    "fixed_ip": null,
    "instance_id": null
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.5.4 Querying Floating IP Addresses (Discarded)

Function

This API is used to query floating IP addresses.

This API has been discarded. Use the API described in "Querying Floating IP Addresses".

URI

GET /v2.1/{project_id}/os-floating-ips

Table 11-79 describes the parameters in the URI.

Table 11-79 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

Table 11-80 describes the response parameters.

Table 11-80 Response parameters

Parameter	Mandatory	Type	Description
floating_ips	Yes	Array of objects	Specifies the floating IP addresses.

Table 11-81 floating_ip objects

Parameter	Mandatory	Type	Description
floating_ip	Yes	Object	Specifies the floating IP address.

Table 11-82 floating_ip attributes

Parameter	Mandatory	Type	Description
fixed_ip	Yes	String	Specifies a private IP address.

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the floating IP address ID in UUID format.
instance_id	Yes	String	Specifies the ID of a bound ECS in UUID format.
ip	Yes	String	Specifies the floating IP address.
pool	Yes	String	Specifies the name of a network resource pool that provides floating IP addresses.

Example Request

```
GET https://{endpoint}/v2.1/e73621affb8f44e1bc01898747ca09d4/os-floating-ips
```

Example Response

```
{
  "floating_ips": [
    {
      "id": "05f71f43-f3c9-47ef-ac8d-9f02aef66418",
      "pool": "external",
      "ip": "10.154.51.235",
      "fixed_ip": "192.168.1.2",
      "instance_id": "8b380f68-5057-4aa2-a33a-170b37218fa8"
    },
    {
      "id": "a25236cf-dd76-4adc-916a-f0b4a24048d3",
      "pool": "external",
      "ip": "10.154.51.237",
      "fixed_ip": null,
      "instance_id": null
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

11.5.5 Querying Details About a Floating IP Address (Discarded)

Function

This API is used to query the details about a floating IP address based on the ID of the floating IP address.

This API has been discarded. Use the API described in "Querying a Floating IP Address".

URI

GET /v2.1/{project_id}/os-floating-ips/{floating_ip_id}

[Table 11-83](#) describes the parameters in the URI.

Table 11-83 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
floating_ip_id	Yes	Specifies the ID of the floating IP address.

Request

None

Response

[Table 11-84](#) describes the response parameters.

Table 11-84 Response parameters

Parameter	Mandatory	Type	Description
floating_ip	Yes	Object	Specifies the floating IP address. For details, see Table 11-85 .

Table 11-85 floating_ip objects

Parameter	Mandatory	Type	Description
fixed_ip	Yes	String	Specifies a private IP address.
id	Yes	String	Specifies the floating IP address ID in UUID format.
instance_id	Yes	String	Specifies the ID of a bound ECS in UUID format.
ip	Yes	String	Specifies the floating IP address.

Parameter	Mandatory	Type	Description
pool	Yes	String	Specifies the name of a network resource pool that provides floating IP addresses.

Example Request

```
GET https://{endpoint}/v2.1/e73621affb8f44e1bc01898747ca09d4/os-floating-ips/05f71f43-f3c9-47ef-ac8d-9f02aef66418
```

Example Response

```
{
  "floating_ip":{
    "id": "05f71f43-f3c9-47ef-ac8d-9f02aef66418",
    "pool": "external",
    "ip": "10.154.51.235",
    "fixed_ip": "192.168.1.2",
    "instance_id": "8b380f68-5057-4aa2-a33a-170b37218fa8"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.5.6 Releasing a Floating IP Address (Discarded)

Function

This API is used to release a floating IP address.

This API has been discarded. Use the API described in "Deleting a Floating IP Address".

URI

```
DELETE /v2.1/{project_id}/os-floating-ips/{floating_ip_id}
```

[Table 11-86](#) describes the parameters in the URI.

Table 11-86 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
floating_ip_id	Yes	Specifies the ID of the floating IP address.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/e73621affb8f44e1bc01898747ca09d4/os-floating-ips/05f71f43-f3c9-47ef-ac8d-9f02aef66418
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

11.5.7 Querying Floating IP Address Pools (Discarded)

Function

This API is used to query floating IP address pools.

This API has been discarded. Use the API described in "Querying Networks".

Constraints

The API parameter is as follows: router:external=True

```
GET /networks?router:external=True //Name in the result is returned.
```

URI

```
GET /v2.1/{project_id}/os-floating-ip-pools
```

[Table 11-87](#) describes the parameters in the URI.

Table 11-87 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

None

Response

[Table 11-88](#) describes the response parameters.

Table 11-88 Response parameters

Parameter	Mandatory	Type	Description
floating_ip_pools	Yes	Array of objects	Specifies the floating IP address pool.
name	Yes	String	Specifies the name of the floating IP address pool.

Example Request

```
GET https://{endpoint}/v2.1/e73621affb8f44e1bc01898747ca09d4/os-floating-ip-pools
```

Example Response

```
{
  "floating_ip_pools": [
    {
      "name": "pool1"
    },
    {
      "name": "pool2"
    }
  ]
}
```

Returned Values

See [Returned Values for General Requests](#).

11.6 Snapshot Management (OpenStack Nova APIs)

11.6.1 Creating a Snapshot (Discarded)

Function

This API is used to create a snapshot for a volume.

This API has been discarded. Use the EVS API "Creating an EVS Snapshot (OpenStack Cinder API v2)".

Constraints

A snapshot name cannot be prefixed with **autobk_snapshot**.

URI

```
POST /v2.1/{project_id}/os-snapshots
```

[Table 11-89](#) describes the parameters in the URI.

Table 11-89 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .

Request

[Table 11-90](#) describes the request parameters.

Table 11-90 Request parameters

Parameter	Mandatory	Type	Description
display_description	No	String	Specifies the snapshot description.
volume_id	Yes	String	Specifies the volume ID.
display_name	No	String	Specifies the name of the EVS snapshot. The value contains a maximum of 255 bytes. NOTE When creating a backup for an EVS disk through VBS, a snapshot will be created and named with prefix autobk_snapshot_ . The EVS console has imposed operation restrictions on snapshots with prefix autobk_snapshot_ . You are advised to not use autobk_snapshot_ as the name prefix for the snapshots you created. Otherwise, the snapshots cannot be used normally.
force	No	Boolean	Specifies whether a snapshot is to be forcibly created. If the value is true , a snapshot can be created for a disk in the in-use state.

Response

[Table 11-91](#) describes the response parameters.

Table 11-91 Response parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the disk snapshot ID in UUID format.
status	Yes	String	Specifies the volume snapshot status.
displayName	No	String	Specifies the volume snapshot name.
displayDescription	No	String	Specifies the volume snapshot description.
createdAt	Yes	String	Specifies the time when the volume snapshot was created.
volumeId	Yes	String	Specifies the disk ID in UUID format for the snapshot.
size	Yes	Integer	Specifies the volume snapshot size.

Example Request

```
POST https://{endpoint}/v2.1/d6c277ba8820452e83df36f33c9fa561/os-snapshots
{
  "snapshot": {
    "display_name": "test",
    "display_description": null,
    "volume_id": "ba5730ea-8621-4ae8-b702-ff0ffc12c209"
  }
}
```

Example Response

```
{
  "snapshot": {
    "createdAt": "2016-05-20T16:54:14.981520",
    "displayDescription": null,
    "id": "b836dc3d-4e10-4ea4-a34c-8f6b0460a583",
    "displayName": "test",
    "size": 1,
    "status": "creating",
    "volumeId": "ba5730ea-8621-4ae8-b702-ff0ffc12c209"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.6.2 Querying Snapshots (Discarded)

Function

This API is used to query information about a volume snapshot.

This API has been discarded. Use the EVS API "Querying Details About an EVS Snapshot (OpenStack Cinder API v2)".

URI

GET /v2.1/{project_id}/os-snapshots/{snapshot_id}

[Table 11-92](#) describes the parameters in the URI.

Table 11-92 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
snapshot_id	Yes	Specifies the volume snapshot ID.

Response

Response parameters

[Table 11-93](#) describes the response parameters.

Table 11-93 Response parameters

Parameter	Type	Description
id	String	Specifies the disk snapshot ID in UUID format.
status	String	Specifies the volume snapshot status.
displayName	String	Specifies the volume snapshot name.
displayDescription	String	Specifies the volume snapshot description.
createdAt	String	Specifies the time when the volume snapshot was created.
volumeId	String	Specifies the disk ID in UUID format for the snapshot.

Parameter	Type	Description
size	Integer	Specifies the volume snapshot size.

Example Request

```
GET https://{endpoint}/v2.1/d6c277ba8820452e83df36f33c9fa561/os-snapshots/b836dc3d-4e10-4ea4-a34c-8f6b0460a583
```

Example Response

```
{
  "snapshot":
  {
    "createdAt": "2016-05-20T16:54:14.981520",
    "displayDescription": null,
    "id": "b836dc3d-4e10-4ea4-a34c-8f6b0460a583",
    "displayName": "test",
    "size": 1,
    "status": "creating",
    "volumeId": "ba5730ea-8621-4ae8-b702-ff0ffc12c209"
  }
}
```

Returned Values

See [Returned Values for General Requests](#).

11.6.3 Deleting a Snapshot (Discarded)

Function

This API is used to delete a volume snapshot.

This API has been discarded. Use the EVS API "Deleting an EVS Snapshot (OpenStack Cinder API v2)".

URI

DELETE /v2.1/{project_id}/os-snapshots/{snapshot_id}

[Table 11-94](#) describes the parameters in the URI.

Table 11-94 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see Obtaining a Project ID .
snapshot_id	Yes	Specifies the volume snapshot ID.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/d6c277ba8820452e83df36f33c9fa561/os-snapshots/  
74bfbbdd-7af5-4ed5-81b2-0aed668441d6
```

Example Response

None

Returned Values

See [Returned Values for General Requests](#).

A Appendix

A.1 HTTP Status Codes

Normal Status Code	Description
200	OK
201	Created
202	Accepted
204	No Content

Error Status Code	Description
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
405	Method Not Allowed
409	Conflict
413	Request Entity Too Large
415	Unsupported Media Type
429	Too Many Requests
500	Internal Server Error
501	Not Implemented
503	Service Unavailable

A.2 Error Codes

Context

- An error code returned by an API does not correspond to one error message. The following table lists only common error messages.
- Most ECS APIs are asynchronous. Some error codes are displayed in the returned messages for task viewing requests. HTTP status codes may not be accurate.
- The ECS service is strongly dependent on other services, such as network and storage. If the reported error messages contain information about ECS-depended services, contact technical support for troubleshooting.
- If the system displays an error code when you perform operations on the management console, see "How Do I Handle Error Messages Displayed on the Management Console?" in the *Elastic Cloud Server User Guide* for troubleshooting.

Error Codes

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0000	Request error. Try again later or contact customer service.	Request error.	Check the request body according to the returned error message.
400	Ecs.0001	Insufficient ECS quota. Contact customer service to increase quota.	The number of ECSs has reached the maximum allowed.	Apply for a higher quota of the corresponding resource according to the returned error message.
400	Ecs.0002	A system exception occurred. Try again later or contact customer service.	Failed to submit the task.	Contact technical support to locate the fault.
403	Ecs.0003	You do not have permission to perform this operation. Contact customer service to obtain permission.	You do not have permission or your balance is insufficient.	Check whether the account balance is insufficient or the account is frozen according to the returned error message.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0004	A system exception occurred. Try again later or contact customer service.	Authentication failed.	For details, see the returned error message or contact technical support.
400	Ecs.0005	Invalid parameter values. Contact customer service.	Invalid parameters.	Check whether the request body is of the correct JSON structure according to the API reference.
400	Ecs.0007	A system exception occurred. Try again later or contact customer service.	Invalid image attributes.	Adjust the specifications or image type.
400	Ecs.0008	A system exception occurred. Try again later or contact customer service.	Invalid flavor attributes.	Contact technical support to check whether the flavor registration is valid.
400	Ecs.0009	Another flavor must be used for resizing.	The flavor is not changed.	Select a flavor different from the current flavor.
400	Ecs.0010	The private IP address is already being used. Select another IP address.	The private IP address is already in use.	Change the port.
400	Ecs.0011	Ensure the password meets the password complexity requirements.	Failed to meet password complexity requirements.	Check the password length and change the password.
400	Ecs.0012	The subnet does not contain enough IP addresses. Release some IP addresses or select a different subnet.	The number of IP addresses in the subnet is insufficient.	Check whether the floating IP addresses of the subnet are used up.
400	Ecs.0013	The current EIP quota limit has been reached. Apply to increase the quota.	Insufficient EIP quota.	Apply for a higher EIP quota because the EIP quota is insufficient.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0014	Incorrect VPC, subnet, or security group parameter values.	Invalid VPC parameters.	Check whether the subnets belong to the same VPC.
400	Ecs.0015	Invalid disk type for this type of ECS. Select a valid disk type and try again.	The disk of this type is not applicable to the ECS.	Check whether the disk type is supported by the flavor.
400	Ecs.0017	The status of the selected disk does not meet the attachment requirements on the ECS. Select an available disk for attaching.	The ECS is not the target one that the system disk or data disk is to be attached.	Check whether the __system_server_id value in disk metadata is the same as the UUID of the ECS to which the system disk or data disk is to be attached.
400	Ecs.0021	Insufficient EVS disk quota. Contact customer service to increase quota.	Insufficient EVS disk quota.	Apply for a higher EVS disk quota.
400	Ecs.0022	Insufficient ECS group quota. Contact customer service to increase quota.	The number of ECSs in the ECS group exceeded the upper limit.	Apply for a higher ECS quota for an ECS group.
400	Ecs.0023	project_id in token mismatches with project_id in url.	Invalid token, or the project ID in the token is different from that in the URL.	Apply for a valid token or check the project ID in the URL.
400	Ecs.0025	EVS is not authorized to obtain KMS keys for encrypting EVS disks.	EVS is not authorized to obtain KMS keys for encrypting EVS disks.	Authorize EVS to obtain KMS keys for encrypting EVS disks.
400	Ecs.0027	The ECSs of this flavor cannot be created.	Private flavor, which cannot be used.	Switch to another flavor.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0028	The ECSs of this flavor cannot be created.	The blacklisted user configured in the flavor is not allowed to use the flavor.	Switch to another flavor.
400	Ecs.0029	The flavor does not exist.	The flavor does not exist or has been abandoned.	Switch to another flavor.
400	Ecs.0030	The ECS has been frozen and does not support specifications modification.	The ECS has been frozen.	Check whether the account has been frozen or contact technical support.
400	Ecs.0031	The image does not exist.	The image does not exist.	Replace the image.
400	Ecs.0032	The image is not in Active state.	The image is not in Active state.	Replace the image.
400	Ecs.0034	The full-ECS backup does not exist or has been deleted.	The full-ECS backup does not exist or has been deleted.	Replace the image.
400	Ecs.0036	The flavor does not support automatic recovery.	The flavor does not support automatic recovery.	Change another flavor.
400	Ecs.0037	The flavor does not support SCSI disks.	The flavor does not support SCSI disks.	Change another flavor or type.
400	Ecs.0038	The subnet does not exist.	The subnet does not exist.	Adjust the network parameter settings.
400	Ecs.0039	The specified IP address does not belong to the subnet.	The specified IP address does not belong to the subnet.	Change the specified private IP address.
400	Ecs.0041	Invalid description field.	Invalid description field.	Modify the service description field.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0042	The number of attached data disks exceeds the maximum allowed limit.	The number of attached data disks exceeds the maximum allowed limit.	Adjust the number of attached data disks.
400	Ecs.0043	The disk type does not exist.	The disk type does not exist.	Change the disk type.
400	Ecs.0044	The disk of this type has been sold out.	The disk of this type has been sold out.	Change the disk type.
400	Ecs.0045	The bandwidth exceeds the maximum allowed limit.	The bandwidth exceeds the maximum size allowed.	Adjust the bandwidth.
400	Ecs.0046	When creating an ECS using an image, ensure that the type of the attached data disk is the same as that required by the image.	The disk type of the ECS is different from that of the snapshot image.	Change the disk type.
400	Ecs.0048	Ensure that the image status is Normal and that the status of the CSBS backup associated with the image is Available or Creating, and try again later.	The full-ECS image is unavailable.	Check the full-ECS image.
400	Ecs.0050	The number of NICs attached to the ECS exceeds the quota.	The number of NICs attached to the ECS exceeds the maximum value allowed.	Adjust the number of NICs.
400	Ecs.0051	Only SCSI disks can be attached to the ECSs of this flavor.	The attached disk is not of SCSI type.	Change the disk type.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0052	Only SCSI system disks can be attached to the ECSs of this flavor.	The attached system disk is not of SCSI type.	Change the system disk type.
400	Ecs.0053	Only SCSI data disks can be attached to the ECSs of this flavor.	The attached data disk is not of SCSI type.	Change the data disk type.
400	Ecs.0057	The disk has already been attached to the ECS and you cannot repeatedly attach it.	The disk has been attached to the ECS.	Attach a new disk to the ECS.
400	Ecs.0058	You do not have permission to use a third-party image to create ECSs.	The providedId of the image does not match the account ID.	Check the account permission and image.
400	Ecs.0062	The flavor does not support the driver mode.	The flavor does not allow settings of the NIC driver type.	Change another flavor.
400	Ecs.0064	The VPC ID in the request is inconsistent with that in the main subnet ID.	Inconsistent VPC ID in the request body from that in the primary NIC.	Adjust the NIC parameter settings.
403	Ecs.0066	This operation cannot be performed because real-name authentication has not been completed.	Restricted due to lack of real-name authentication.	Check whether the account is restricted due to lack of real-name authentication.
403	Ecs.0067	Insufficient account balance.	Restricted due to insufficient balance.	Check whether the account is restricted due to insufficient balance.

HTTP Status Code	Error Code	Error Message	Description	Solution
403	Ecs.0068	This operation cannot be performed by partners.	Restricted due to a non-partner.	Check whether the account is restricted due to a non-partner.
403	Ecs.0069	You have not associated a payment method with your account.	Restricted due to incomplete payment information.	Check whether the payment information of the account is complete.
403	Ecs.0070	Insufficient budget. Contact the enterprise administrator and request for a budget increase.	Restricted because account budget of the enterprise department is insufficient.	Check whether the budget of the enterprise department account is sufficient.
403	Ecs.0071	This operation cannot be performed because your account has been suspended.	Restricted due to a malicious account.	Check whether the account is malicious.
400	Ecs.0073	The system disk is being backed up. Wait until the execution is complete and try again.	The system disk is being backed up.	You are not allowed to delete a system disk that is being backed up.
400	Ecs.0074	Window images do not support external users.	External users are not allowed to create Windows ECSs.	External users, including and non-third-party users, are not allowed to purchase Windows images.
400	Ecs.0075	Partners only support Windows images.	Partners can purchase only Windows images.	Purchase only Windows images.
400	Ecs.0085	The server does not have the interface.	The ECS does not have the NIC.	Replace a NIC.
400	Ecs.0086	The interface is not the primary interface.	The NIC is not the primary NIC.	Replace a NIC.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0090	Image license type is BYOL, the BYOL feature is not supported at this time.	BYOL image features are not supported in this region.	Change the image or contact technical support.
400	Ecs.0100	The ECS status does not meet requirements. Make the ECS in the required status and try again.	The ECS status does not meet requirements.	The ECS in the current state does not support this operation. Try again later.
400	Ecs.0101	The system disk is currently unresponsive. Try again later or contact customer service.	Abnormal system disk status.	For details, contact technical support.
400	Ecs.0102	The data disk is currently unresponsive. Try again later or contact customer service.	The system disk status does not allow the disk to be detached.	Check the system disk status.
400	Ecs.0103	The disk can be attached to a server only if it exists and the state must be in the available. Make sure the disk state is available and try again.	The disk is unavailable.	Check the disk status or contact technical support to change the disk status.
400	Ecs.0104	The number of EVS disks that can be attached to the ECS exceeds the maximum number allowed. Decrease the number of EVS disks to be attached and try again.	Insufficient ECS disk quota for attaching more disks.	Adjust the number of attached disks.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0105	No system disk found. Attach the system disk to the ECS and try again.	Failed to query the ECS system disk.	Check whether the ECS has a system disk attached.
400	Ecs.0106	A network exception occurred. Try again later or contact customer service.	Abnormal network status.	Contact technical support to locate the fault.
403	Ecs.0110	Contact the main account to obtain permission.	Operations are prohibited on the client due to permissions.	You do not have the permission to perform such an operation. Check token permissions. For details, see the error message returned by the API.
400	Ecs.0111	The EVS disk has been detached from the ECS. Refresh the disk list and check the disk.	The disk is not in the attachment list.	Check whether the selected disk has been attached to the ECS, or replace the disk.
404	Ecs.0114	The ECS does not exist.	The ECS cannot be detected.	Check whether the ECS has been created.
400	Ecs.0118	The number of ECSs exceeds the maximum allowed limit.	The number of tasks in a batch is greater than the upper limit.	Check the number of ECSs in the batch.
400	Ecs.0119	An encrypted disk with an unavailable key cannot be attached to an ECS.	An ECS cannot be attached with an encrypted disk with a disabled key.	Change the key status.
400	Ecs.0121	The disk cannot be attached to the ECS because the disk and the ECS are in different failure domains.	Failed to attach the disk because the ECS and the disk are in different failure domains.	Select a disk that is in the same failure domain as that of the target ECS.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0201	Failed to create the NIC. Try again later or contact customer service.	Failed to create the NIC.	For details, see the returned error message or contact technical support.
400	Ecs.0202	Failed to create the system disk. Try again later or contact customer service.	Failed to create the system disk.	For details, see the returned error message or contact technical support.
400	Ecs.0203	Failed to create the data disk. Try again later or contact customer service.	Failed to create the data disk.	For details, see the returned error message or contact technical support.
400	Ecs.0204	Failed to create the ECS. Try again later or contact customer service.	Failed to create the ECS.	For details, see the returned error message or contact technical support.
400	Ecs.0205	Failed to attach the data disk. Try again later or contact customer service.	Failed to attach the data disk.	For details, see the returned error message or contact technical support.
400	Ecs.0207	Failed to modify the ECS specifications. Try again later or contact customer service.	Failed to modify ECS specifications.	For details, see the returned error message or contact technical support.
400	Ecs.0208	A system exception occurred. Try again later or contact customer service.	Failed to update the image metadata.	For details, see the returned error message or contact technical support.
400	Ecs.0209	Failed to modify the ECS specifications. Try again or contact customer service.	Failed to confirm the ECS specifications modification.	For details, see the returned error message or contact technical support.
400	Ecs.0210	A system exception occurred. Try again later or contact customer service.	Failed to assign the floating IP address.	For details, see the returned error message or contact technical support.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0211	Failed to create the NIC. Try again later or contact customer service.	Failed to create the NIC QoS.	For details, see the returned error message or contact technical support.
400	Ecs.0212	Failed to assign the private IP address. Try again later or contact customer service.	Failed to allocate the private IP address.	For details, contact technical support.
400	Ecs.0213	Failed to update the port attributes. Try again later or contact customer service.	Failed to update the port attributes.	For details, see the returned error message or contact technical support.
400	Ecs.0214	Failed to create the network. Try again later or contact customer service.	Failed to create the network.	For details, see the returned error message or contact technical support.
400	Ecs.0216	Failed to create the subnet. Try again later or contact customer service.	Failed to create the subnet.	For details, see the returned error message or contact technical support.
400	Ecs.0217	Failed to attach the NIC. Try again later or contact customer service.	Failed to attach the NIC.	For details, see the returned error message or contact technical support.
400	Ecs.0219	Failed to create the ECS. Try again later or contact customer service.	Failed to create the ECS.	For details, see the returned error message or contact technical support.
400	Ecs.0221	Cold migration from a dedicated host to the same dedicated host is not supported.	Failed to migrate the ECS.	For details, see the returned error message or contact technical support.
400	Ecs.0226	Failed to start.	Failed to start the ECS.	For details, see the returned error message or contact technical support.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0227	Failed to reboot.	Failed to restart the ECS.	Modify according to the returned error message or contact technical support.
400	Ecs.0301	Failed to query the ECS. Try again later or contact customer service.	Failed to query the ECS.	For details, see the returned error message or contact technical support.
400	Ecs.0302	Failed to query the ECS quota of the tenant. Try again later or contact customer service.	Failed to query the ECS quota of the tenant.	For details, see the returned error message or contact technical support.
400	Ecs.0303	Failed to query the ECS specifications. Try again later or contact customer service.	Failed to query the flavor.	For details, see the returned error message or contact technical support.
400	Ecs.0304	Failed to query the image. Try again later or contact customer service.	Failed to query the image.	Contact technical support to check whether the image has been correctly registered or to check other causes.
400	Ecs.0306	Failed to query the backup. Try again later or contact customer service.	Failed to query the backup.	For details, see the returned error message or contact technical support.
400	Ecs.0307	Failed to query the port. Try again later or contact customer service.	Failed to query the port.	For details, see the returned error message or contact technical support.
400	Ecs.0308	Failed to query the ECS quota of the tenant. Try again later or contact customer service.	Failed to query the ECS quota of the tenant.	For details, see the returned error message or contact technical support.
400	Ecs.0309	Failed to create the NIC. Try again later or contact customer service.	Failed to query the NIC QoS.	For details, see the returned error message or contact technical support.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0310	A system exception occurred. Try again later or contact customer service.	Failed to view the network information.	For details, see the returned error message or contact technical support.
400	Ecs.0311	Failed to obtain the disk type. Try again later or contact customer service.	Failed to query the disk type.	For details, see the returned error message or contact technical support.
400	Ecs.0313	ECS group query failed.	Failed to query the ECS group.	For details, see the returned error message or contact technical support.
400	Ecs.0314	The key pair does not exist. Refresh the key pair list and check key pair	Failed to obtain the key pair.	For details, see the returned error message or contact technical support.
400	Ecs.0315	Failed to call the nova API to query the auto recovery status.	Failed to obtain the automatic recovery status.	For details, see the returned error message or contact technical support.
400	Ecs.0319	Insufficient resources for this flavor. Try another flavor.	Insufficient flavor capacity.	Apply for expanding the flavor capacity.
400	Ecs.0320	AZ query failed.	Failed to obtain AZs.	For details, see the returned error message or contact technical support.
400	Ecs.0321	Console logs query failed.	Failed to query ECS console logs.	For details, see the returned error message or contact technical support.
400	Ecs.0322	Subnet query failed.	Failed to query details of the subnet.	For details, see the returned error message or contact technical support.
400	Ecs.0323	Failed to query the NIC attached to the ECS.	Failed to query the NIC attachment to an ECS.	For details, see the returned error message or contact technical support.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0401	Failed to release the port. Try again later or contact customer service.	Failed to undo the operation performed on the port.	For details, see the returned error message or contact technical support.
400	Ecs.0402	Failed to release the system disk. Try again later or contact customer service.	Failed to undo the operation performed on the system disk.	For details, see the returned error message or contact technical support.
400	Ecs.0403	Failed to release the ECS. Try again later or contact customer service.	Failed to undo the operation performed on the ECS.	Contact technical support to locate the fault.
400	Ecs.0405	Failed to release the data disk. Try again later or contact customer service.	Failed to undo the operation performed on the data disk.	For details, see the returned error message or contact technical support.
400	Ecs.0501	Failed to delete the ECS. Try again later or contact customer service.	Failed to delete the ECS.	Try again later.
400	Ecs.0502	Failed to delete the private IP address. Try again later or contact customer service.	Failed to delete the private IP address.	For details, see the returned error message or contact technical support.
400	Ecs.0503	Failed to obtain the system disk. Try again later or contact customer service.	Failed to query the system disk.	For details, see the returned error message or contact technical support.
400	Ecs.0507	Failed to delete the NIC. Try again later or contact customer service.	Failed to delete the NIC.	Check the NIC type.
501	Ecs.0603	The commands are being executed. Try again later.	Other commands are being executed. Try again 1 minute later.	Try again 1 minute later.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0605	ECS locked.	The ECS is locked.	Check whether the ECS is locked. If so, unlock it.
400	Ecs.0611	Batch operation failed.	Requesting for a batch operation failed.	Rectify the fault based on the returned error information and submit the request again.
404	Ecs.0614	The ECS does not exist.	The ECS cannot be detected.	Check whether the ECS exists.
500	Ecs.0615	The thread list is empty.	An error has occurred in the request from an ECS.	An internal system error occurred. Contact technical support to locate the fault.
400	Ecs.0616	Failed to update the ECS name.	Failed to modify the ECS.	Try again later or contact technical support.
400	Ecs.0617	Failed to modify attribute. Please try again later or contact customer service.	Failed to modify the attributes of the disk attached to an ECS.	For details, see the returned error message or contact technical support.
400	Ecs.0618	Failed to change the IP address of the ECS NIC.	Failed to change the IP address of the ECS NIC.	For details, see the returned error message or contact technical support.
400	Ecs.0707	The product has not been registered.	The product does not exist.	Contact technical support to locate the fault.
400	Ecs.0803	When modifying the specifications of an ECS created on a DeH, specify the DeH.	Modifying specifications of an ECS deployed on a DeH requires the DeH ID.	Modify the request body for specifications modification.
400	Ecs.0810	The ECS flavor is the same as the target flavor.	The target specifications are the same as the current ECS specifications.	Switch to another flavor.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0811	Install the required drivers on the ECS and then change Xen to KVM.	The flavor cannot be switched from Xen to KVM.	Install a driver script.
400	Ecs.0904	UEFI images cannot be used to create Xen ECSs.	UEFI images cannot be used to create Xen ECSs.	Switch to another flavor.
400	Ecs.0905	The number of tags exceeds the maximum allowed limit.	The number of tags exceeds the maximum number allowed.	Decrease the number of tags.
400	Ecs.0906	Failed to comply with tag character set specifications.	Invalid tag attribute.	Create a tag again.
400	Ecs.0907	Invalid tag character set.	Invalid tag character set.	Create a tag again.
400	Ecs.0908	The tag key cannot be duplicate.	Duplicate tag key.	Create a tag again.
400	Ecs.0909	The flavor does not support the disk type.	The flavor does not support the disk type.	Change the flavor or disk type.
400	Ecs.0910	Invalid NIC settings for creating a HANA ECS.	Invalid NIC parameters for creating a HANA ECS.	Adjust the NIC parameter settings.
400	Ecs.0911	Invalid dedicated storage type of the disk.	Invalid dedicated storage type of the disk.	Modify parameter settings for the dedicated storage type.
400	Ecs.0912	Invalid disk encryption key.	Invalid disk encryption attribute.	Modify parameter settings for the disk encryption attribute.
400	Ecs.0913	The number of ECSs to be created exceeds the maximum allowed limit	The number of ECSs to be created exceeds the maximum allowed limit.	Decrease the number of ECSs to be created.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.0914	The length of the ECS name exceeds the maximum allowed limit.	The length of the ECS name exceeds the maximum allowed limit.	Change the ECS name.
400	Ecs.0915	The length of the ECS name exceeds the maximum allowed limit.	The ECS name contains invalid characters.	Change the ECS name.
400	Ecs.0919	The NIC has been attached to another instance.	The port does not allow attaching.	Change the port.
400	Ecs.1000	A system exception occurred. Try again later or contact customer service.	Failed to call the Nova API.	Internal calling error. Try again later or contact technical support.
400	Ecs.1001	A system exception occurred. Try again later or contact customer service.	OpenStack access error.	The ECS is abnormal due to an OpenStack exception. Contact technical support.
400	Ecs.1002	A system exception occurred. Try again later or contact customer service.	OpenStack access timed out.	If you are switching a VPC, attaching or detaching a NIC or disk, or changing or reinstalling an OS, stop the process first and try again. If the retry timed out, contact technical support.
400	Ecs.1100	A system exception occurred. Try again later or contact customer service.	Failed to access IAM.	For details, see the returned error message or contact technical support.
400	Ecs.1200	A system exception occurred. Try again later or contact customer service.	Failed to access the VPC.	For details, see the returned error message or contact technical support.
400	Ecs.1201	A system exception occurred. Try again later or contact customer service.	VPC access timed out.	The task timed out. For details, contact technical support.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Ecs.1300	A system exception occurred. Try again later or contact customer service.	EVS access timed out.	For details, see the returned error message or contact technical support.
403	Pdp.0001	Policy does not allow %s to be performed.	API authentication failed.	Add permissions on IAM. For details, see API permissions.
202	Comm on.0024	exceeds flow over limit	Limited by traffic control.	The number of concurrent requests has exceeded the upper limit. Try again later.
400	Comm on.0002	The request body cannot be left blank.	Empty request body.	Check the request body.
400	Comm on.0011	Failed to query system tasks.	Invalid job ID.	Check whether the source of the job ID is correct.
400	Comm on.0018	The project ID in the URL is different from that in the token.	Invalid token, or the project ID in the token is different from that in the URL.	Check whether the tenant token is correct.
400	Comm on.0020	A system exception occurred. Try again later or contact customer service.	Failed to retry the task.	Contact technical support.
400	Comm on.0021	Subjob fails.	An error has occurred in job query.	Try again later or contact technical support.
400	Comm on.0022	Mission fails.	An error has occurred in job submission.	Contact technical support.
400	Comm on.0999	The system was broken, exit.	Task terminated.	Contact technical support.
400	Comm on.0025	Query job Error because %s.	An error has occurred in job query.	Try again later or contact technical support.

HTTP Status Code	Error Code	Error Message	Description	Solution
400	Common.0026	Fail to get Region Info	An error occurred in AZ query.	Try again later or contact technical support.
401	Common.0013	Invalid token.	Invalid token.	Check whether the tenant token is correct.
500	Common.0001	A system exception occurred. Try again later or contact customer service.	A system exception occurred.	Contact technical support.
429	Common.1503	Api flow control Error because %s.	Limited by API traffic control.	Too many APIs are being executed. Try again later.

A.3 ECS Statuses

An ECS can be in one of the following statuses specified in ECS APIs:

- **status**: specifies an ECS status, which is generated by **OS-EXT-STS:vm_state** and **OS-EXT-STS:task_state**.
- **OS-EXT-STS:vm_state**: indicates that the ECS is in a stable state after an operation is performed. This is an extended attribute.
- **OS-EXT-STS:task_state**: indicates an intermediate status in which the ECS is processing an operation performed on it. This is an extended attribute.

Table A-1 Statuses

Status	Description
BUILD	The ECS has been created but is not running.
REBOOT	The ECS is being restarted.
HARD_REBOOT	The ECS is being forcibly restarted.
REBUILD	The ECS is being rebuilt.
MIGRATING	The ECS is being live migrated.
RESIZE	The ECS has received a specifications modification request and has started to perform the modification.
ACTIVE	The ECS is running properly.
SHUTOFF	The ECS has been properly stopped.

Status	Description
REVERT_RESIZE	The ECS is rolling back resizing.
VERIFY_RESIZE	The ECS is verifying the modified configuration.
ERROR	An error has occurred on the ECS.
DELETED	The ECS has been deleted.
SHELVED	The ECS boot from an image is shelved.
SHELVED_OFFLOADED	The ECS boot from a volume is shelved.
UNKNOWN	The ECS status is unknown.

Table A-2 OS-EXT-STS:vm_state statuses

Status	Description
building	The ECS has been created but is not running.
active	The ECS is running properly.
stopped	The ECS has been properly stopped.
resized	The ECS specifications have been modified.
error	An error has occurred on the ECS.
deleted	The ECS has been deleted.
shelved	The ECS boot from an image is shelved.
shelved_offloaded	The ECS boot from a volume is shelved.

Table A-3 OS-EXT-STS:task_state statuses

Status	Description
scheduling	The ECS is being created.
block_device_mapping	The ECS is being created, and disks are being prepared for the ECS.
networking	The ECS is being created, and network resources are being prepared for the ECS.
spawning	The ECS is being created.
rebooting	The ECS is being restarted.

Status	Description
reboot_pending	A restarting command has been issued to an ECS, and the ECS is to be restarted.
reboot_started	The ECS is being restarted.
rebooting_hard	The ECS is being forcibly restarted.
reboot_pending_hard	A forcible restarting command has been issued to an ECS, and the ECS is to be restarted.
reboot_started_hard	The ECS is being forcibly restarted.
rebuilding	The ECS is being rebuilt.
rebuild_block_device_mapping	The ECS is being rebuilt, and disks are being prepared for the ECS.
rebuild_spawnning	The ECS is being rebuilt.
migrating	The ECS is being live migrated.
resize_prep	The ECS specifications are to be modified, and resources are being prepared for the ECS.
resize_migrating	The specifications of the ECS are being modified, and it is being migrated.
resize_migrated	The specifications of the ECS are being modified, and it has been migrated.
resize_finish	The specifications of the ECS are being modified.
resize_reverting	The specifications modification of the ECS is being rolled back.
powering-off	The ECS is stopped.
powering-on	The ECS is being started.
deleting	The ECS is being deleted.
shelving	The ECS boot from an image is being shelved.
shelving_offloading	The ECS boot from a volume is being shelved.
shelving_image_pending_upload	A shelving image is pending uploaded.
shelving_image_uploading	A shelving image is pending uploaded.
unshelving	The ECS is being unshelved.

Table A-4 Mapping between statuses

vm_state	task_state	status
building	scheduling block_device_mapping networking spawning null	BUILD
active	rebooting reboot_pending reboot_started	REBOOT
	rebooting_hard reboot_pending_hard reboot_started_hard	HARD_REBOOT
	rebuilding rebuild_block_device_mapping rebuild_spawning	REBUILD
	migrating	MIGRATING
	powering-off deleting null	ACTIVE
stopped	resize_prep resize_migrating resize_migrated resize_finish	RESIZE
	rebuilding rebuild_block_device_mapping rebuild_spawning	REBUILD
	powering-on deleting null	SHUTOFF
resized	resize_reverting	REVERT_RESIZE
	null	VERIFY_RESIZE
error	rebuilding rebuild_block_device_mapping rebuild_spawning	REBUILD

vm_state	task_state	status
	deleting null	ERROR
deleted	null	DELETED
shelved	shelving shelving_image_pending_upload shelving_image_uploading unshelving null	SHELVED
shelved_offloaded	shelving_offloading unshelving null	SHELVED_OFFLOADED

 **NOTE**

If the status is not included in [Table A-4](#), the status is UNKNOWN.