

Elastic Cloud Server

API Reference

Issue 14
Date 2021-04-08



Copyright © Huawei Technologies Co., Ltd. 2021. 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 Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

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

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 API Calling.....	1
1.3 Endpoints.....	1
1.4 Constraints.....	1
1.5 Concepts.....	2
1.6 Selecting an API Type or Version.....	3
1.7 Querying Data in Pages.....	8
2 API Overview.....	9
3 Calling APIs.....	12
3.1 Making an API Request.....	12
3.2 Authentication.....	16
3.3 Response.....	18
4 APIs (Recommended).....	20
4.1 Lifecycle Management.....	20
4.1.1 Creating ECSs.....	20
4.1.2 Creating an ECS (Pay-per-Use).....	38
4.1.3 Deleting ECSs.....	51
4.1.4 Querying Details About an ECS.....	53
4.1.5 Querying Details About ECSs.....	59
4.1.6 Modifying an ECS.....	66
4.2 Status Management.....	71
4.2.1 Reinstalling an ECS OS (Using an Image with Cloud-Init Installed).....	72
4.2.2 Changing an ECS OS (Using an Image with Cloud-Init Installed).....	75
4.2.3 Reinstalling an ECS OS (Using an Image Without Cloud-Init Installed).....	79
4.2.4 Changing an ECS OS (Using an Image Without Cloud-Init Installed).....	82
4.2.5 Querying Automatic Recovery of an ECS.....	84
4.2.6 Managing Automatic Recovery of an ECS.....	86
4.2.7 Cold Migrating an ECS.....	87
4.2.8 Obtaining the VNC Login Address.....	89
4.2.9 Modifying the Specifications of an ECS.....	91
4.2.10 Modifying the Specifications of an ECS (Pay-per-Use).....	93

4.2.11 Adding an ECS to the Monitoring List.....	95
4.3 Batch Operations.....	96
4.3.1 Starting ECSs in a Batch.....	96
4.3.2 Restarting ECSs in a Batch.....	98
4.3.3 Stopping ECSs in a Batch.....	100
4.3.4 Modifying ECSs in a Batch.....	101
4.3.5 Resetting the Passwords for Logging In to ECSs in a Batch.....	103
4.3.6 Attaching a Specified Shared EVS Disk to Multiple ECSs.....	106
4.4 Flavor Management.....	109
4.4.1 Querying Details About Flavors and Extended Flavor Information.....	109
4.4.2 Querying the Target ECS Flavors to Which a Flavor Can Be Changed.....	116
4.4.3 Querying the Target Flavors to Which an ECS Flavor Can Be Changed (Discarded).....	120
4.5 NIC Management.....	125
4.5.1 Adding NICs to an ECS in a Batch.....	125
4.5.2 Deleting NICs from an ECS in a Batch.....	127
4.5.3 Binding a Virtual IP Address to an ECS NIC.....	128
4.5.4 Unbinding a Virtual IP Address from an ECS NIC.....	130
4.5.5 Query NICs of an ECS.....	132
4.6 Disk Management.....	134
4.6.1 Querying a Single Disk Attached to an ECS.....	134
4.6.2 Querying Disk Attachments of an ECS.....	136
4.6.3 Querying Information About Disks Attached to an ECS.....	138
4.6.4 Attaching a Disk to an ECS.....	140
4.6.5 Detaching an EVS Disk from an ECS.....	142
4.6.6 Querying Disk Attachment of an ECS (Discarded).....	143
4.6.7 Querying a Single Disk Attached to an ECS (Discarded).....	145
4.7 Metadata Management.....	147
4.7.1 Updating ECS Metadata.....	147
4.7.2 Deleting Specified ECS Metadata.....	149
4.8 Tenant Quota Management.....	149
4.8.1 Querying Tenant Quotas.....	150
4.9 Task Status Management.....	152
4.9.1 Querying Task Execution Status.....	152
4.10 Tag Management.....	156
4.10.1 Tag Types.....	156
4.10.2 Adding Tags to an ECS in a Batch.....	156
4.10.3 Deleting Tags from an ECS in a Batch.....	158
4.10.4 Querying Project Tags.....	160
4.10.5 Querying Tags of an ECS.....	162
4.10.6 Querying Tags of an ECS (Discarded).....	163
4.11 Password Management.....	164
4.11.1 Querying Whether One-Click Password Reset Is Supported.....	165

4.11.2 Resetting the Password for Logging In to an ECS with a Few Clicks.....	166
4.11.3 Retrieving the Password for Logging In to a Windows ECS.....	168
4.11.4 Deleting the Password for Logging In to a Windows ECS.....	169
4.11.5 Resetting the Password for Logging In to an ECS with a Few Clicks (Discarded).....	170
4.12 ECS Group Management.....	172
4.12.1 Creating an ECS Group.....	172
4.12.2 Deleting an ECS Group.....	175
4.12.3 Querying ECS Groups.....	176
4.12.4 Querying Details About an ECS Group.....	178
5 OpenStack Nova APIs.....	180
5.1 API Version Query.....	180
5.1.1 Querying All API Versions.....	180
5.1.2 Querying a Specified API Version.....	183
5.2 Lifecycle Management.....	187
5.2.1 Creating an ECS.....	187
5.2.2 Modifying ECS Information.....	202
5.2.3 Deleting an ECS.....	207
5.2.4 Querying ECSs.....	208
5.2.5 Querying Details About ECSs.....	212
5.2.6 Querying Details About ECSs.....	222
5.3 Status Management.....	231
5.3.1 Starting an ECS.....	231
5.3.2 Restarting an ECS.....	232
5.3.3 Stopping an ECS.....	233
5.3.4 Locking an ECS.....	234
5.3.5 Unlocking an ECS.....	235
5.3.6 Creating an Image Using an ECS.....	236
5.3.7 Modifying the Specifications of an ECS.....	239
5.3.8 Confirming ECS Specifications Modification.....	240
5.3.9 Rolling Back ECS Specifications Modification.....	241
5.4 Network Management.....	242
5.4.1 Querying Networks.....	242
5.4.2 Querying the Networks of a Specified ECS.....	244
5.4.3 Querying the Specified Network of an ECS.....	246
5.5 Security Group Management.....	248
5.5.1 Adding a Security Group.....	248
5.5.2 Deleting a Security Group.....	249
5.5.3 Querying Security Groups for a Specified ECS.....	250
5.6 Flavor Management.....	254
5.6.1 Querying ECS Flavors.....	254
5.6.2 Querying Details About ECS Flavors.....	257
5.6.3 Querying Details About an ECS Flavor.....	260

5.6.4 Querying the extra_specs Value for an ECS Flavor.....	263
5.7 NIC Management.....	264
5.7.1 Querying ECS NICs.....	264
5.7.2 Querying NICs of an ECS.....	265
5.7.3 Adding a NIC to an ECS.....	267
5.7.4 Deleting a NIC from an ECS.....	270
5.8 Disk Management.....	271
5.8.1 Querying Disks Attached to an ECS.....	271
5.8.2 Querying a Disk Attached to an ECS.....	272
5.8.3 Attaching a Disk to an ECS.....	274
5.8.4 Detaching a Disk from an ECS.....	276
5.9 Metadata Management.....	279
5.9.1 Updating ECS Metadata.....	279
5.9.2 Configuring ECS Metadata.....	281
5.9.3 Deleting Specified ECS Metadata.....	283
5.9.4 Querying ECS Metadata.....	284
5.9.5 Obtaining ECS Metadata with a Specified Key.....	285
5.9.6 Modifying ECS Metadata with a Specified Key.....	286
5.10 Tenant Quota Management.....	287
5.10.1 Querying Tenant Quota Limits.....	288
5.10.2 Querying Tenant Quotas.....	290
5.10.3 Querying Default Quotas.....	293
5.11 Key and Password Management.....	295
5.11.1 Querying SSH Key Pairs.....	295
5.11.2 Querying a Specified SSH Key Pair.....	296
5.11.3 Creating and Importing an SSH Key Pair.....	298
5.11.4 Deleting an SSH Key Pair.....	302
5.11.5 Retrieving the Password for Logging In to a Windows ECS.....	303
5.11.6 Deleting the Password for Logging In to a Windows ECS.....	304
5.12 ECS Group Management.....	304
5.12.1 Creating an ECS Group.....	305
5.12.2 Querying ECS Groups.....	308
5.12.3 Querying Details About an ECS Group.....	310
5.12.4 Deleting an ECS Group.....	311
5.13 ECS Operation Management.....	312
5.13.1 Querying Operations on an ECS.....	312
5.13.2 Querying ECS Operations by Request ID.....	314
5.14 ECS Console Management.....	316
5.14.1 Obtaining ECS Management Console Logs.....	316
5.14.2 Obtaining a VNC-based Remote Login Address (Microversion 2.6 or Later).....	318
5.15 AZ.....	320
5.15.1 Querying AZs.....	320

5.16 Tag Management.....	321
5.16.1 Querying Tags of an ECS.....	322
5.16.2 Adding Tags to an ECS.....	323
5.16.3 Deleting Tags from an ECS.....	324
5.16.4 Adding a Tag to an ECS.....	325
5.16.5 Querying a Specified Tag for an ECS.....	326
5.16.6 Deleting a Specified Tag from an ECS.....	327
5.17 Historical Versions.....	328
6 Application Examples.....	329
6.1 Example 1: Creating an ECS.....	329
6.2 Example 2: Querying ECSs.....	336
6.3 Example 3: Modifying ECS Specifications.....	340
6.4 Example 4: Attaching a Disk to an ECS.....	342
6.5 Example 5: Attaching a NIC to an ECS.....	344
6.6 Example 6: Querying the EIP Associated with an ECS.....	348
7 Data Structure.....	351
7.1 Data Structure for Creating ECSs.....	351
7.2 Data Structure for Querying Details About ECSs.....	363
7.3 Data Structure for Query Details About Specifications.....	367
8 Permissions Policies and Supported Actions.....	372
8.1 Lifecycle Management.....	372
8.2 ECS Status Management.....	376
8.3 Batch Operations.....	377
8.4 Network Management.....	378
8.5 Image Management.....	378
8.6 Security Group Management.....	379
8.7 Specifications Query.....	380
8.8 NIC Management.....	381
8.9 Disk Management.....	383
8.10 Metadata Management.....	385
8.11 Tenant Quota Management.....	386
8.12 SSH Key Management.....	387
8.13 Password Management.....	387
8.14 Floating IP Address Management.....	388
8.15 ECS Group Management.....	389
8.16 ECS Management Through Console.....	390
8.17 AZ Management.....	390
8.18 Tag Management.....	390
9 Common Parameters.....	392
9.1 Returned Values for General Requests.....	392
9.2 Error Code Description.....	393

9.3 HTTP Status Code.....	410
9.4 Obtaining a Project ID.....	411
9.5 Task Request Result.....	412
9.5.1 Responses (Task).....	412
9.5.2 Returned Values.....	413
9.6 Batch Task Request.....	414
9.6.1 Responses (Batch Operation).....	414
10 Out-of-Date APIs.....	417
10.1 Image Management (OpenStack Nova APIs).....	417
10.1.1 Querying Images (Discarded).....	417
10.1.2 Querying Image Details (Discarded).....	421
10.1.3 Querying Details About a Specified Image (Discarded).....	425
10.1.4 Querying the Metadata of a Specified Image (Discarded).....	427
10.1.5 Deleting an Image (Discarded).....	429
10.2 Security Group Management (OpenStack Nova APIs).....	430
10.2.1 Querying Security Groups (Discarded).....	430
10.2.2 Creating a Security Group (Discarded).....	434
10.2.3 Querying Details About a Security Group (Discarded).....	435
10.2.4 Updating a Security Group (Discarded).....	438
10.2.5 Deleting a Security Group (Discarded).....	442
10.2.6 Creating a Security Group Rule (Discarded).....	443
10.2.7 Deleting a Security Group Rule (Discarded).....	446
10.3 Disk Management (OpenStack Nova APIs).....	447
10.3.1 Querying Brief Information About Disks (Discarded).....	447
10.3.2 Querying Detailed Information About Disks (Discarded).....	449
10.3.3 Querying Information About a Disk (Discarded).....	451
10.3.4 Creating a Disk (Discarded).....	453
10.3.5 Deleting a Disk (Discarded).....	456
10.4 Floating IP Address Management (OpenStack Nova APIs).....	457
10.4.1 Binding a Floating IP Address (Discarded).....	457
10.4.2 Unbinding a Floating IP Address (Discarded).....	458
10.4.3 Assigning a Floating IP Address (Discarded).....	460
10.4.4 Querying Floating IP Addresses (Discarded).....	461
10.4.5 Querying Details About a Floating IP Address (Discarded).....	463
10.4.6 Releasing a Floating IP Address (Discarded).....	465
10.4.7 Querying Floating IP Address Pools (Discarded).....	466
10.5 Snapshot Management (OpenStack Nova APIs).....	467
10.5.1 Creating a Snapshot (Discarded).....	467
10.5.2 Querying Snapshots (Discarded).....	469
10.5.3 Deleting a Snapshot (Discarded).....	470
A Appendix.....	472
A.1 ECS Statuses.....	472

A.2 Network APIs.....	475
B Change History.....	476

1 Before You Start

1.1 Overview

Welcome to *Elastic Cloud Server API Reference*. An Elastic Cloud Server (ECS) is a computing server that consists of CPUs, memory, image, and EVS disks allowing on-demand allocation and elastic scaling. ECS integrates the Virtual Private Cloud (VPC), virtual firewall, and multi-data-copy capabilities to construct an efficient, reliable, and secure computing environment. This ensures stable and uninterrupted operation of services. 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 [2 API Overview](#).

If you plan to access ECSs through an API, ensure that you are familiar with ECS concepts. For details, see [Service Overview](#).

1.2 API Calling

ECSs support Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see [3 Calling APIs](#).

1.3 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](#).

1.4 Constraints

- The number of ECSs that you can create is determined by your quota. To view or increase the quota, see [Quota Adjustment](#).
- For more constraints, see API description.

1.5 Concepts

- **Account**

An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity, which should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.
- **User**

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

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

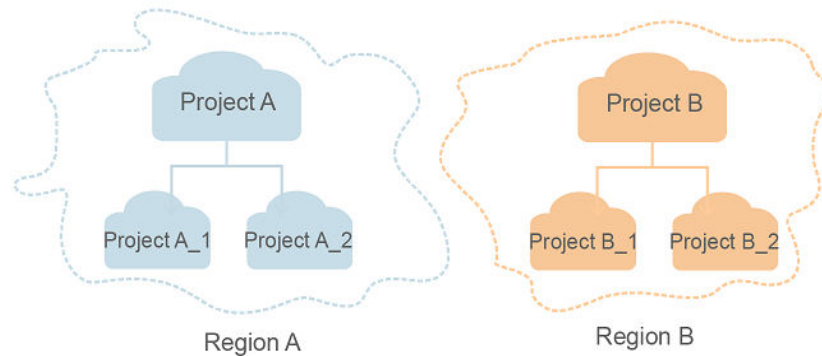
Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region.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

For details, see [Region and AZ](#).
- **AZ**

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

A project corresponds to a region. Default projects are defined 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



- **Enterprise project**
Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated. An enterprise project can contain resources of multiple regions, and resources can be added to or removed from enterprise projects.
For details about enterprise projects and about how to obtain enterprise project IDs, see [Enterprise Management User Guide](#).

1.6 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. ECS APIs can be of V1 or V1.1. 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 [5.1.1 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 [1.3 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 [1.3 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"
}
]
```

1.7 Querying Data in Pages

Nova APIs allow users to query data using search criteria. The **limit** and **marker** parameters are added to the URL of the list request to enable the system to display query results in pages. The query results are displayed by creation time (**create_time**) of the records in ascending order. If the creation time is not provided, the results are displayed by object ID in ascending order.

Parameter	Type	Mandatory	Description
limit	String	No	Restricts the number of records displayed on each page. If the limit value exceeds the maximum number of records that can be displayed on each page, error code 403 will be returned.
marker	String	No	Indicates the ID of the last record on the previous page. If the marker value is invalid, error code 400 will be returned.

next ref in the response indicates the URL of the next page.

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; reinstall or change the ECS OS; automatically recover ECSs.
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.
Metadata management	Update ECS metadata and delete specified ECS metadata.
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.

Type	Description
Tag management	Create, delete, or query ECS D2 tags. Tags can be added or deleted in a batch.
Password management	Reset the password for logging in to an ECS with a few clicks.
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.

Type	Description
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.
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.
AZ	Show AZs.
Tag management	Create, delete, or query ECS D1 tags.

3 Calling APIs

3.1 Making an API Request

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

Request URI

A request URI is in the following format:

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

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

Table 3-1 URI parameter description

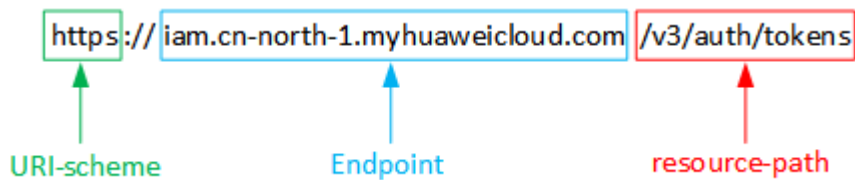
Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from Regions and Endpoints . For example, the endpoint of IAM in the CN North-Beijing1 region is iam.cn-north-1.myhuaweicloud.com .
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is /v3/auth/tokens .

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

For example, to obtain an IAM token in the **CN North-Beijing1** region, obtain the endpoint of IAM (**iam.cn-north-1.myhuaweicloud.com**) for this region and the **resource-path** (**/v3/auth/tokens**) in the URI of the API used to **obtain a user token**. Then, construct the URI as follows:

`https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens`

Figure 3-1 Example URI



NOTE

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

Request Methods

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

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Same as GET except that the server must return only the response header.

Method	Description
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

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

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

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

Common request header fields are as follows.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of <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

Parameter	Description	Mandatory	Example Value
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in 9.4 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
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: MIIPAgYJKoZlhvcNAQcCo...ggg1BBIINPXsidG9rZ

NOTE

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

For more details, see "Authentication Using AK/SK" in [3.2 Authentication](#).

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

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

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

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

In the case of the API used to [obtain a user token](#), the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*,

domainname, ******* (login password), and *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://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

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

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

3.2 Authentication

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

- Token-based authentication: Requests are authenticated using a token.
- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair. AK/SK-based authentication is recommended because it is more secure than token-based authentication.

Token-based 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. When you call the API, set **auth.scope** in the request body to **project**.

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

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

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

AK/SK-based Authentication

NOTE

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

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

NOTE

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

3.3 Response

Status Code

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

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see [9.3 HTTP Status Code](#).

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

Response Header

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

[Figure 3-2](#) shows the response header fields for the API used to [obtain a user token](#). The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-2 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
→ MIIVXQVJKoZIhvcNAQcCoIIYJCCEoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOansiZXhwaXJlc19hdCI6ijlwMTktMDItMTNUMC
fj3KIs6YgKnpVNRbW2eZ5eb78SZOkqjACgkklqO1wi4JlGzrpd18LGXK5tdfq4lqHCYb8P4NaY0NyejcAgzJVeFYtLWT1GSO0zxKZmlQHQj82HBqHdglZO9fuEbL5dMhdavj+33wEI
xHRCe9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXI1jipPEGA270g1FruooL6jqglFkNPQuFSOUB+uSsttVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUUpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==
x-xss-protection → 1; mode=block;
```

(Optional) Response Body

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

The following is part of the response body for the API used to [obtain a user token](#).

```
{
  "token": {
```

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

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

```
{  
  "error_msg": "The format of message is error",  
  "error_code": "AS.0001"  
}
```

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

4 APIs (Recommended)

4.1 Lifecycle Management

4.1.1 Creating ECSs

Function

This API is used to create one or more ECSs.

The V1.1 API supports all functions (see [4.1.2 Creating an ECS \(Pay-per-Use\)](#)) provided by the V1 API. Additionally, the V1.1 API supports the creation of yearly/monthly ECSs.

This is an asynchronous API. After the ECS creation request is issued, the system will return **job_id**. The ECS creation is still in progress. Therefore, you need to call the API described in [4.9.1 Querying Task Execution Status](#) to obtain the task status. When the status changes to **SUCCESS**, the ECS has been created.

Logging in to an ECS can be authenticated using either a key pair or password. The login using a key pair is more secure than using a password. Therefore, key pair authentication is recommended.

- 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. For details, see [Table 4-2](#).

- 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 4-2](#). 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 4-2](#).

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.

Constraints

- Ensure that your account has sufficient balance because this API does not support coupons. If the account balance is insufficient, a pending order will be generated.

URI

- URI format
POST /v1.1/{project_id}/cloudservers
- Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

Table 4-1 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS information. For details, see Table 4-2 .

Parameter	Mandatory	Type	Description
dry_run	No	Boolean	<p>Specifies whether to check the request and create the ECS.</p> <ul style="list-style-type: none"> • true: indicates that only 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: indicates that the request is sent and the ECS will be created if the check result is as expected.

Table 4-2 Parameters for creating an ECS

Parameter	Mandatory	Type	Description
imageRef	Yes	String	<p>Specifies the ID of the system image used for creating ECSs. The ID is in Universally Unique Identifier (UUID) format.</p> <p>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>.</p>
flavorRef	Yes	String	<p>Specifies the flavor ID of the ECS to be created.</p> <p>For details about the flavors that have been released, see "Instances" in <i>Elastic Cloud Server User Guide</i>.</p>

Parameter	Mandatory	Type	Description
name	Yes	String	<p>Specifies the ECS name.</p> <p>For details, see How Can I Set Sequential ECS Names When Creating Multiple ECSs?</p> <p>A name must comply with the following rules:</p> <ul style="list-style-type: none"> • The parameter value consists of 1 to 64 characters, including letters, digits, underscores (_), hyphens (-), 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, letters (case sensitive), and hyphens (-). Underscores (_) are converted into hyphens (-) by default.</p>

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected during the ECS 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 <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>lyEgL2Jpbi9iYXNoDQplY2hvlHVzZXJfdGVzd-CAmZ3Q7Jmd0OyAvaG9tZS91c2VyLnR4dA==</pre> Windows <pre>cmVtIGNtZA0KZWNobyAxMTEgJmd0OyBjOlxhYWEudHh0</pre>
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> <p>Password complexity requirements:</p> <ul style="list-style-type: none"> Consists of 8 to 26 characters. Contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@\$%^_+=+[{ }];,./?). 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.

Parameter	Mandatory	Type	Description
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 creating API (5.11.3 Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (5.11.1 Querying SSH Key Pairs).</p> <p>Note:</p> <p>If chargeMode in the extendparam parameter of a created ECS is set to prePaid, which indicates that the ECS is billed in yearly/monthly payments, the key_name parameter must be used with the metadata parameter. For details, see metadata Field Description for Creating ECSs and example request 1.</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 4-3.</p> <p>Note:</p> <ul style="list-style-type: none"> • The network of the primary NIC must belong to the VPC specified by vpcid. When you create NICs, the first NIC specified is the primary NIC. • A maximum of 12 NICs can be attached to an ECS by default. • The maximum number of NICs varies depending on ECS specifications. For details, see ECS Specifications.

Parameter	Mandatory	Type	Description
publicip	No	Object	<p>Specifies the EIP bound to the ECS, which can be configured in one of the following ways:</p> <ul style="list-style-type: none">• Do not use: In such a case, this field is unavailable.• Automatically assign an EIP. You need to specify the EIP.• Use existing one. You need to specify an existing EIP. <p>For details, see publicip Field Description.</p>
count	No	Integer	<p>Specifies the number of ECSs to be created.</p> <p>Note:</p> <ul style="list-style-type: none">• If this parameter is not specified, the default value is 1.• If chargingMode in the extendparam parameter is set to postPaid, the ECS is billed in pay-per-use payments, and a tenant can create a maximum of 500 ECSs.• If chargingMode in the extendparam parameter is set to prePaid, the ECS is billed in yearly/monthly payments, and a tenant can create a maximum of 100 ECSs. A maximum of 400 resources can be purchased at a time. For example, a purchased ECS includes mandatory resources, such as one cloud server and one system disk, and other optional resources, such as data disks, EIP, and bandwidth. All of these are included in 400 resources. The system will report an error when the number of resources exceeds 400.
root_volume	Yes	Object	<p>Specifies ECS system disk configurations. The system disk and data disk created during the creation of a yearly/monthly ECS are also in yearly/monthly payments, and the period of the disks is the same as that of the ECS.</p> <p>For details, see Table 4-4.</p>

Parameter	Mandatory	Type	Description
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 4-5 .
security_groups	No	Array of objects	Specifies ECS security groups. Constraints: If this parameter is left blank, the default security group is bound to the ECS by default. For details, see security_groups Field Description .
availability_zone	No	String	Specifies the name of the AZ where the ECS is located. See Regions and Endpoints .
extendparam	No	Object	Specifies the ECS supplementary information. For details, see Table 7-8 .
metadata	No	Object	Specifies the ECS metadata. You can use metadata to customize key-value pairs. NOTE <ul style="list-style-type: none"> • 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, digits, hyphens (-), underscores (_), colons (:), and decimal points (.). • A metadata value consists of a maximum of 255 characters. For details about reserved key-value pairs, see Table 7-10 .
os:scheduler_hints	No	Object	Schedules ECSs, for example, by configuring an ECS group. For details, see Table 7-11 .

Parameter	Mandatory	Type	Description
tags	No	Array of strings	<p>Specifies the tags of an ECS.</p> <p>A tag is in the format of "key.value", where the maximum lengths of key and value are 36 and 43 characters, respectively.</p> <p>When adding a tag to an ECS, ensure that the tag complies with the following requirements:</p> <ul style="list-style-type: none"> • The key of the tag can contain only uppercase letters, lowercase letters, digits, underscores (_), and hyphens (-). • The value of the tag can contain only uppercase letters, lowercase letters, digits, underscores (_), hyphens (-), and periods (. <p>NOTE</p> <ul style="list-style-type: none"> • When you create ECSs, one ECS supports up to 10 tags. • The functions of the server_tags field newly added to the public cloud are the same as those of the tags field, but the former supports more keys and values. Therefore, the server_tags field is recommended.

Table 4-3 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> <p>You can obtain the parameter value by calling a VPC API for Querying Subnet Details.</p>

Parameter	Mandatory	Type	Description
ip_address	No	String	<p>Specifies the IP address of the NIC used by the ECS. The value is an IPv4 address.</p> <p>Constraints:</p> <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.

Table 4-4 root_volume field description

Parameter	Mandatory	Type	Description
volumetype	Yes	String	<p>Specifies the ECS system disk type, which must be one of available disk types.</p> <p>Currently, the value can be SSD, GPSSD, or SAS.</p> <ul style="list-style-type: none"> • SSD: specifies the ultra-high I/O disk type. • GPSSD: specifies the general purpose SSD disk type. • SAS: specifies the high I/O disk type. <p>If the specified disk type is not available in the AZ, the disk will fail to create.</p> <p>NOTE</p> <ul style="list-style-type: none"> • For details about disk types, see Disk Types and Disk Performance.

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>
extendparam	No	Object	<p>Provides the disk information.</p> <p>For details, see extendparam Field Description for Creating Disks.</p>
hw:passthrough	No	Boolean	<p>Pay attention to this parameter if your ECS is SDI-compliant. If the value of this parameter is true, the created disk is of SCSI type.</p> <p>NOTE This parameter is of boolean type. If a non-boolean character is imported, the parameter value is set to false.</p>

Table 4-5 data_volumes field description

Parameter	Mandatory	Type	Description
volumetype	Yes	String	<p>Specifies the type of the ECS data disk, which must be one of available disk types. Currently, the value can be SSD, GPSSD, or SAS.</p> <ul style="list-style-type: none"> • SSD: specifies the ultra-high I/O disk type. • GPSSD: specifies the general purpose SSD disk type. • SAS: specifies the high I/O disk type. <p>If the specified disk type is not available in the AZ, the disk will fail to create.</p> <p>NOTE</p> <ul style="list-style-type: none"> • For details about disk types, see Disk Types and Disk Performance.
size	Yes	Integer	<p>Specifies the data disk size, in GB. The value ranges from 10 to 32768.</p> <p>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.</p>
shareable	No	Boolean	<p>Specifies whether the disk is shared. The value can be true (specifies a shared disk) or false (a common EVS disk).</p> <p>NOTE This field has been discarded. Use multiattach.</p>
multiattach	No	Boolean	<p>Specifies the shared disk information.</p> <ul style="list-style-type: none"> • true: indicates that the created disk is a shared disk. • false: indicates that the created disk is a common EVS disk. <p>NOTE The shareable field is not used any more. If both shareable and multiattach must be used, ensure that the values of the two fields are the same. If this parameter is not specified, common EVS disks are created by default.</p>

Parameter	Mandatory	Type	Description
hw:passthrough	No	Boolean	<p>Indicates whether the data volume uses a SCSI lock.</p> <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 this parameter does not appear, the disk device type is VBD. <p>NOTE This parameter is of boolean type. If a non-boolean character is imported, the parameter value is set to false.</p>
extendparam	No	Object	<p>Provides the disk information. For details, see Table 7-6.</p>
data_image_id	No	String	<p>Specifies ID of the data image. The value is in UUID format.</p> <p>If data disks are created using a data disk image, this parameter is mandatory and it does not support metadata.</p>
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

Table 4-6 Response parameters

Parameter	Type	Description
job_id	String	Specifies the returned task ID after delivering the task. You can query the task progress using this ID. For details how to query the execution status of the task based on the task ID, see 4.9 Task Status Management .
order_id	String	Specifies the order ID. This parameter is returned for the creation of a yearly/monthly ECS.
serverIds	Array of strings	Specifies ECS IDs. NOTE The details about an ECS are obtained by ECS ID. If the system returns a 404 error, the ECS is being created, or creating the ECS failed.

For details about abnormal responses, see [9.5.1 Responses \(Task\)](#).

Example Request

- Example URL request
POST `https://{endpoint}/v1.1/{project_id}/cloudservers`
- Example request 1 (creating a yearly/monthly ECS that is logged in using a key pair)

```
{
  "server": {
    "availability_zone": "az1-dc1", //AZ name
    "name": "newservers",
    "imageRef": "5ef3a512-1c65-418e-8764-a4413c2f9277",
    "root_volume": {
      "volumetype": "SSD"
    },
    "data_volumes": [
      {
        "volumetype": "SSD",
        "size": 100
      },
      {
        "volumetype": "SSD",
        "size": 100,
        "multiattach": true,
        "hw:passthrough": true
      }
    ],
    "flavorRef": "s2.small.1",
    "vpcid": "2a6f4aa6-d93e-45f5-a8cb-b030dbf8cd68",
    "security_groups": [
      {
        "id": "6242ef48-4d35-49c8-8711-a6e54902e44a"
      }
    ],
    "nics": [
      {
        "subnet_id": "ef039b60-6a14-42d1-963b-687b627fea08"
      }
    ]
  }
}
```

```
    ],
    "publicip": {
      "eip": {
        "iptype": "5_sbgp",
        "bandwidth": {
          "size": 1,
          "sharetype": "PER"
        }
      }
    },
    "key_name": "id_rsa",
    "count": 1,
    "metadata": {
      "op_svc_userid": "f79791beca3c48159ac2553fff22e166"
    },
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,
      "isAutoRenew": "true",
      "isAutoPay": "true",
      "enterprise_project_id": "f8e0ecc8-3825-4ee8-9596-fb4258ffdccb"
    },
    "os:scheduler_hints": {
      "group": "cdbbffe-ef18-47b4-a5c8-f61a984c0ecc"
    }
  }
}
```

- Example request 2 (creating a yearly/monthly ECS that is logged in using a password)

```
{
  "server": {
    "availability_zone": "az1-dc1", //AZ name
    "name": "newservers",
    "adminPass": "P@ssw0rd123",
    "imageRef": "9b04ad7e-6d97-40bf-9d62-57873382eab0",
    "root_volume": {
      "volumetype": "SSD"
    },
    "data_volumes": [
      {
        "volumetype": "SSD",
        "size": 100
      },
      {
        "volumetype": "SSD",
        "size": 100,
        "multiattach": true,
        "hw.passthrough": true
      }
    ],
    "flavorRef": "s2.small.1",
    "vpcid": "2a6f4aa6-d93e-45f5-a8cb-b030dbf8cd68",
    "security_groups": [
      {
        "id": "6242ef48-4d35-49c8-8711-a6e54902e44a"
      }
    ],
    "nics": [
      {
        "subnet_id": "ef039b60-6a14-42d1-963b-687b627fea08"
      }
    ],
    "publicip": {
      "eip": {
        "iptype": "5_sbgp",
        "bandwidth": {
          "size": 1,

```

```

        "sharetype": "PER"
    }
}
},
"key_name": "",
"count": 1,
"metadata": {},
"extendparam": {
    "chargingMode": "prePaid",
    "periodType": "month",
    "periodNum": 1,
    "isAutoRenew": "true",
    "isAutoPay": "true",
    "enterprise_project_id": "f8e0ecc8-3825-4ee8-9596-fb4258ffdcbb"
},
"os:scheduler_hints": {
    "group": "cddbfffe-ef18-47b4-a5c8-f61a984c0ecc"
}
}
}
}

```

- Example request 3 (creating a yearly/monthly ECS with a pay-per-use EIP bound)

```

{
  "server": {
    "availability_zone": "az1-dc1", //AZ name
    "name": "newservers",
    "imageRef": "5ef3a512-1c65-418e-8764-a4413c2f9277",
    "root_volume": {
      "volumetype": "SSD"
    },
    "data_volumes": [
      {
        "volumetype": "SSD",
        "size": 100
      },
      {
        "volumetype": "SSD",
        "size": 100,
        "multiattach": true,
        "hw:passthrough": true
      }
    ],
    "flavorRef": "s2.small.1",
    "vpcid": "2a6f4aa6-d93e-45f5-a8cb-b030dbf8cd68",
    "security_groups": [
      {
        "id": "6242ef48-4d35-49c8-8711-a6e54902e44a"
      }
    ],
    "nics": [
      {
        "subnet_id": "ef039b60-6a14-42d1-963b-687b627fea08"
      }
    ],
    "publicip": {
      "eip": {
        "iptype": "5_sbgp",
        "bandwidth": {
          "size": 1,
          "sharetype": "PER",
          "chargemode": "traffic"
        },
        "extendparam": {
          "chargingMode": "postPaid"
        }
      }
    },
    "key_name": "id_rsa",
    "count": 1,
  }
}

```

```
"metadata": {
  "op_svc_userid": "f79791beca3c48159ac2553fff22e166"
},
"extendparam": {
  "chargingMode": "prePaid",
  "periodType": "month",
  "periodNum": 1,
  "isAutoRenew": "true",
  "isAutoPay": "true",
  "enterprise_project_id": "f8e0ecc8-3825-4ee8-9596-fb4258ffdccb"
},
"os:scheduler_hints": {
  "group": "cdbbffe-ef18-47b4-a5c8-f61a984c0ecc"
}
}
```

- Example request 4 (creating a yearly/monthly ECS with an EIP using a shared bandwidth bound)

```
{
  "server": {
    "availability_zone": "az1-dc1", //AZ name
    "name": "newserver",
    "imageRef": "5ef3a512-1c65-418e-8764-a4413c2f9277",
    "root_volume": {
      "volumetype": "SSD"
    },
    "data_volumes": [
      {
        "volumetype": "SSD",
        "size": 100
      },
      {
        "volumetype": "SSD",
        "size": 100,
        "multiattach": true,
        "hw:passthrough": true
      }
    ],
    "flavorRef": "s2.small.1",
    "vpcid": "2a6f4aa6-d93e-45f5-a8cb-b030dbf8cd68",
    "security_groups": [
      {
        "id": "6242ef48-4d35-49c8-8711-a6e54902e44a"
      }
    ],
    "nics": [
      {
        "subnet_id": "ef039b60-6a14-42d1-963b-687b627fea08"
      }
    ],
    "publicip": {
      "eip": {
        "iptype": "5_sbgp",
        "bandwidth": {
          "id": "a0d4b26f-699d-49a0-bcc8-6f707a925abf",
          "sharetype": "WHOLE"
        }
      }
    },
    "key_name": "id_rsa",
    "count": 1,
    "metadata": {
      "op_svc_userid": "f79791beca3c48159ac2553fff22e166",
      "agency_name": "test"
    },
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,

```

```
    "isAutoRenew": "true",
    "isAutoPay": "true",
    "enterprise_project_id": "f8e0ecc8-3825-4ee8-9596-fb4258ffdccb"
  },
  "os.scheduler_hints": {
    "group": "cddbfffe-ef18-47b4-a5c8-f61a984c0ecc"
  }
}
```

- Example request 6 (pre-verification request body)

```
{
  "dry_run": true
  "server": {
    "availability_zone": "az1-dc1",
    "name": "server",
    "imageRef": "ff49b1f1-3e3e-4913-89c6-a026041661e8",
    "root_volume": {
      "volumetype": "SSD"
    },
    "data_volumes": [],
    "flavorRef": "s2.large.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"
  }
}
```

Example Response

```
{
  "job_id": "ff808082739334d80173943ec9b42130",
  "order_id": "CS2007281506xxxx",
  "serverIds": [
    "fe0528f0-5b1c-4c8c-9adf-e5d5047b8c17",
    "679854ae-a50d-40c9-8132-b19bf3a306a1"
  ]
}
```

Or

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

Or

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

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.1.2 Creating an ECS (Pay-per-Use)

Function

This API is used to create one or more ECSs billed in pay-per-use mode.

This is an asynchronous API. After the ECS creation request is issued, the system will return **job_id**. The ECS creation is still in progress. Therefore, you need to call the API described in [4.9.1 Querying Task Execution Status](#) to obtain the task status. When the status changes to **SUCCESS**, the ECS has been created.

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 4-9](#). 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 4-9](#).

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 4-7](#) describes the parameters in the URI.

Table 4-7 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

Request parameters

[Table 4-8](#) describes the request parameters.

Table 4-8 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS information. For details, see Table 4-9 .
dry_run	No	Boolean	Specifies whether to check the request and create the ECS. <ul style="list-style-type: none">● true: indicates that only 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: indicates that the request is sent and the ECS will be created if the check result is as expected.

Table 4-9 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 "Instances" in <i>Elastic Cloud Server User Guide</i> .
name	Yes	String	Specifies the ECS name. For details, see How Can I Set Sequential ECS Names When Creating Multiple ECSs? Value requirements: <ul style="list-style-type: none">• Consists of 1 to 64 characters, including letters, digits, underscores (_), hyphens (-), 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, letters (case sensitive), and hyphens (-). Underscores (_) are converted into hyphens (-) by default.

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected during the ECS creation process. 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 <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>lyEgL2Jpbi9iYXNoDQplY2hvIHVzZXJfdGVzd-CAmZ3Q7Jmd0OyAvaG9tZS91c2VyLnR4dA==</pre> Windows <pre>cmVtIGNtZA0KZWNobyAxMTEgJmd0OyBjOlxhYWUudHh0</pre>
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> <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.

Parameter	Mandatory	Type	Description
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 creating API (5.11.3 Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (5.11.1 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 4-10.</p> <p>Constraints:</p> <ul style="list-style-type: none">• The network of the primary NIC must belong to the VPC specified by vpcid. When you create NICs, the first NIC specified is the primary NIC.• 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.• The maximum number of NICs varies depending on ECS specifications. For details, see ECS Specifications.
publicip	No	Object	<p>Specifies the EIP of the ECS, which can be configured in the following ways:</p> <ul style="list-style-type: none">• Do not use: In such a case, this field is unavailable.• Automatically assign: The system will automatically assign an EIP to your ECS.• Use existing: You need to specify an existing EIP for your ECS. <p>For details, see Table 7-1.</p>

Parameter	Mandatory	Type	Description
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 4-11 .
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 4-12 .
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 7-2 .
availability_zone	No	String	Specifies the name of the AZ where the ECS is located. See Regions and Endpoints. 5.15.1 Querying AZs .
extendparam	No	Object	Provides the supplementary information about the ECS to be created. For details, see Table 7-7 .

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"> • 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, 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 7-10.</p>
os:scheduler_hints	No	Object	<p>Schedules ECSs, for example, by configuring an ECS group.</p> <p>For details, see Table 7-11.</p>
tags	No	Array of strings	<p>Specifies ECS tags.</p> <p>A tag is in the format of "key.value", where the maximum lengths of key and value are 36 and 43 characters, respectively.</p> <p>When adding a tag to an ECS, ensure that the tag complies with the following requirements:</p> <p>NOTE</p> <ul style="list-style-type: none"> • When you create ECSs, one ECS supports up to 10 tags.
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 4-10 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> <p>You can obtain the parameter value by calling a VPC API for Querying Subnet Details.</p>
ip_address	No	String	<p>Specifies the IP address of the NIC used by the ECS. The value is an IPv4 address.</p> <p>Constraints:</p> <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.

Table 4-11 root_volume field description

Parameter	Mandatory	Type	Description
volumetype	Yes	String	<p>Specifies the ECS system disk type, which must be one of available disk types.</p> <p>Currently, the value can be SSD, GPSSD, or SAS.</p> <ul style="list-style-type: none">• SSD: specifies the ultra-high I/O disk type.• GPSSD: specifies the general purpose SSD disk type.• SAS: specifies the high I/O disk type. <p>If the specified disk type is not available in the AZ, the disk will fail to create.</p> <p>NOTE</p> <ul style="list-style-type: none">• For details about disk types, see Disk Types and Disk Performance.

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>
extendparam	No	Object	<p>Provides the disk information.</p> <p>For details, see extendparam Field Description for Creating Disks.</p>
hw:passthrough	No	Boolean	<p>Pay attention to this parameter if your ECS is SDI-compliant. If the value of this parameter is true, the created disk is of SCSI type.</p> <p>NOTE This parameter is of boolean type. If a non-boolean character is imported, the parameter value is set to false.</p>

Table 4-12 data_volumes field description

Parameter	Mandatory	Type	Description
volumetype	Yes	String	<p>Specifies the type of the ECS data disk, which must be one of available disk types. Currently, the value can be SSD, GPSSD, or SAS.</p> <ul style="list-style-type: none"> • SSD: specifies the ultra-high I/O disk type. • GPSSD: specifies the general purpose SSD disk type. • SAS: specifies the high I/O disk type. <p>If the specified disk type is not available in the AZ, the disk will fail to create.</p> <p>NOTE</p> <ul style="list-style-type: none"> • For details about disk types, see Disk Types and Disk Performance.
size	Yes	Integer	<p>Specifies the data disk size, in GB. The value ranges from 10 to 32768.</p> <p>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.</p>
shareable	No	Boolean	<p>Specifies whether the disk is shared. The value can be true (specifies a shared disk) or false (a common EVS disk).</p> <p>NOTE This field has been discarded. Use multiattach.</p>
multiattach	No	Boolean	<p>Specifies the shared disk information.</p> <ul style="list-style-type: none"> • true: indicates that the created disk is a shared disk. • false: indicates that the created disk is a common EVS disk. <p>NOTE The shareable field is not used any more. If both shareable and multiattach must be used, ensure that the values of the two fields are the same. If this parameter is not specified, common EVS disks are created by default.</p>

Parameter	Mandatory	Type	Description
hw:passthrough	No	Boolean	<p>Indicates whether the data volume uses a SCSI lock.</p> <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 this parameter does not appear, the disk device type is VBD. <p>NOTE This parameter is of boolean type. If a non-boolean character is imported, the parameter value is set to false.</p>
extendparam	No	Object	<p>Provides the disk information. For details, see Table 7-6.</p>
data_image_id	No	String	<p>Specifies ID of the data image. The value is in UUID format.</p> <p>If data disks are created using a data disk image, this parameter is mandatory and it does not support metadata.</p>
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 4.9 Task Status Management.</p>

For details about abnormal responses, see [9.5.1 Responses \(Task\)](#).

Example Request

The public cloud platform provides various ECS types. The flavor name/ID varies depending on ECS types and specifications. When you use APIs to create ECSs with different specifications, the request bodies are the same. You only need to change the parameter values in the following request example based on the parameters described in [Request](#).

- Example URL request

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

- An ECS with flavor ID **s3.xlarge.2** is to be created, where the image ID is **1189efbf-d48b-46ad-a823-94b942e2a000**, disk type is **SSD**, and VPC ID is **0dae26c9-9a70-4392-93f3-87d53115d171**. An example request is as follows:

```
{
  "server": {
    "availability_zone": "az1-dc1",
    "name": "newserver",
    "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"
      }
    ],
    "publicip": {
      "eip": {
        "iptype": "5_bgp",
        "bandwidth": {
          "size": 10,
          "sharetype": "PER"
        }
      }
    },
    "key_name": "sshkey-123",
    "count": 1,
    "server_tags": [
      {
        "key": "key1",
        "value": "value1"
      }
    ],
    "metadata": {
      "op_svc_userid": "8ea65f4099ba412883e2a0da72b96873",
      "agency_name": "test"
    }
  }
}
```

- An example pre-verification request body is as follows:

```
{
  "dry_run": true
  "server": {
    "availability_zone": "az1-dc1",
    "name": "newsrver",
    "imageRef": "1189efbf-d48b-46ad-a823-94b942e2a000",
    "root_volume": {
      "volumetype": "SSD"
    },
    "data_volumes": [ ],
    "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": "privatelp [%s] is not in this subnet [%s]",
    "code": "Ecs.0005",
    "details": [
      {
        "code": "Ecs.0039"
      }
    ]
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.1.3 Deleting ECSs

Function

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

You can delete a single ECS or multiple ECSs in a batch. A maximum of 1000 ECSs can be deleted in a batch.

URI

POST /v1/{project_id}/cloudservers/delete

[Table 4-13](#) describes the parameters in the URI.

Table 4-13 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 4-14](#) describes the request parameters.

Table 4-14 Request parameters

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies the ECSs to be deleted. For details, see Table 4-15 .
delete_publicip	No	Boolean	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 IP address. The value can be true or false . The default value is false . <ul style="list-style-type: none">true: indicates to delete the EIP bound to the ECS when deleting the ECS.false: indicates only to unbind the EIP bound to the ECS when deleting the ECS.

Parameter	Mandatory	Type	Description
delete_volume	No	Boolean	<p>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.</p> <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 4-15 servers field description

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

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

Example request

```
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": "70a599e0-31e7-49b7-b260-868f441e862b"
}
```

Or

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

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.1.4 Querying Details About an ECS

Function

This API is used to query details about an ECS.

The information can be queried includes the ECS billing mode and the ECS frozen status.

URI

GET /v1/{project_id}/cloudservers/{server_id}

[Table 4-16](#) describes the parameters in the URI.

Table 4-16 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-17](#) describes the response parameters.

Table 4-17 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 4-18 .

Table 4-18 server field description

Parameter	Type	Description
status	String	Specifies the ECS status. Options: ACTIVE, BUILD, ERROR, HARD_REBOOT, MIGRATING, REBOOT, REBUILD, RESIZE, REVERT_RESIZE, SHUTOFF, and VERIFY_RESIZE For details, see A.1 ECS Statuses .
updated	String	Specifies the time when the ECS was updated last time. The time is in the format of "2019-05-22T03:30:52Z".
hostId	String	Specifies the ID of the host where the ECS is deployed.
OS-EXT-SRV-ATTR:host	String	Specifies the name of the host on which the ECS is deployed.
addresses	Object	Specifies the network attribute of the ECS. <ul style="list-style-type: none">• key indicates the network name, for example, demo_net.• value indicates the detailed network information. For details, see Table 7-14 .
key_name	String	Specifies the key pair that is used to authenticate an ECS.
image	Object	Specifies the ECS image. For details, see Table 7-21 .
OS-EXT-STS:task_state	String	Specifies the ECS task status. This is an extended attribute. For details, see A.1 ECS Statuses .
OS-EXT-STS:vm_state	String	Specifies the ECS task status. This is an extended attribute. For details, see A.1 ECS Statuses .
OS-EXT-SRV-ATTR:instance_name	String	Specifies the ECS alias. This is an extended attribute.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the name of the host on which the ECS is deployed. This is an extended attribute.
flavor	Object	Specifies the ECS flavor. For details, see Table 7-15 .

Parameter	Type	Description
id	String	Specifies the ECS ID in UUID format.
security_groups	Array of objects	Specifies the security groups of the ECS. For details, see Table 7-16 .
OS-EXT-AZ:availability_zone	String	Specifies the AZ of an ECS. This is an extended attribute.
user_id	String	Specifies the ID of the user for creating the ECS. The value is in UUID format.
name	String	Specifies the ECS name.
created	String	Specifies the time when the ECS was created. The time is in the format of "2019-05-22T03:19:19Z".
tenant_id	String	Specifies the ID of the tenant to which the ECS belongs, which is the project ID in UUID format.
OS-DCF:diskConfig	String	Specifies the disk configuration type. This is an extended attribute. Options: <ul style="list-style-type: none">● MANUAL: The image space is not expanded.● AUTO: The image space of the system disk will be expanded to be as same as the flavor.
accessIPv4	String	Reserved
accessIPv6	String	Reserved
fault	Object	Specifies the cause of the ECS fault. For details, see Table 7-17 .
progress	Integer	Specifies the ECS creation progress. The value ranges from 0 to 100 .
OS-EXT-STS:power_state	Integer	Specifies the power status of the ECS. This is an extended attribute. Options: <ul style="list-style-type: none">● 0: NOSTATE● 1: RUNNING● 4: SHUTDOWN
config_drive	String	Specifies the configuration driver.

Parameter	Type	Description
metadata	Object	Specifies the ECS metadata. For details, see Table 7-19 . NOTE Metadata includes system default fields and the fields set by users.
OS-SRV-USG:launched_at	String	Specifies the time when the ECS was started. The time is in the format of "2019-05-22T03:23:59.000000".
OS-SRV-USG:terminated_at	String	Specifies the time when the ECS was deleted. The time is in the format of "2019-05-22T03:23:59.000000".
os-extended-volumes:volumes_attached	Array of objects	Specifies the disks attached to an ECS. For details, see Table 7-18 .
description	String	Describes the ECS.
host_status	String	Specifies the status of the host accommodating the ECS. <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: The ECS does not have host information.
OS-EXT-SRV-ATTR:hostname	String	Specifies the host name of the ECS.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the ID reserved for the ECSs to be created in a batch. You can use this ID to obtain all the ECSs created in the batch.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the sequence in which ECSs start if the ECSs are created in a batch. The value ranges from 0 to the number of ECSs created in the batch.
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.
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.

Parameter	Type	Description
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the ECS system disk, such as /dev/vda.
OS-EXT-SRV-ATTR:user_data	String	Specifies the user data (information after encoding) configured during ECS creation.
locked	Boolean	Specifies whether an ECS is locked. <ul style="list-style-type: none">● true: The ECS is locked.● false: The ECS is not locked.
tags	Array of strings	Specifies ECS tags.
os:scheduler_hints	Object	Specifies the ECS scheduling information. For details, see Table 7-12 .
sys_tags	Array of objects	Specifies ECS system tags. For details, see Table 7-20 .

Example Request

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

Example Response

```
{
  "server": {
    "id": "4f4b3dfa-eb70-47cf-a60a-998a53bd598a",
    "name": "ecs-2ecf",
    "addresses": {
      "0431c5e5-bc94-4a44-8263-15da2a642435": [{
        "version": "4",
        "addr": "192.168.1.99",
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:df:18:6d",
        "OS-EXT-IPS:port_id": "23037c18-027a-44e5-b6b9-f8d8f113fe02",
        "OS-EXT-IPS:type": "fixed"
      }]
    },
    "flavor": {
      "disk": "0",
      "vcpus": "1",
      "ram": "1024",
      "id": "s3.small.1",
      "name": "s3.small.1"
    },
    "accessIPv4": "",
    "accessIPv6": "",
    "status": "ACTIVE",
    "progress": 0,
    "hostId": "c7145889b2e3202cd295ceddb1742ff8941b827b586861fd0acedf64",
    "updated": "2018-09-13T07:06:51Z",
    "created": "2018-09-13T07:03:44Z",
    "image": {
      "id": "1ce5800a-e487-4c1b-b264-3353a39e2b4b"
    },
    "metadata": {
      "metering.order_id": "CS1809131459IGC24",
      "metering.image_id": "c71b64e7-4767-4406-afde-2c7c7ac2242c",

```

```
"metering.imagetype": "gold",
"metering.resourcespeccode": "s3.small.1.linux",
"image_name": "HEC_Public_Cloudinit_Oracle_Linux_7.4_64bit_40G",
"metering.resourcetype": "1",
"metering.product_id": "00301-117024-0--0",
"cascaed.instance_extrainfo": "pcibridge:2",
"os_bit": "64",
"vpc_id": "0431c5e5-bc94-4a44-8263-15da2a642435",
"os_type": "Linux",
"charging_mode": "1"
},
"tags": [],
"description": "",
"locked": false,
"config_drive": "",
"tenant_id": "ff2eb406effc455aba53174463eb9322",
"user_id": "0bc5e11f91dd48849bb03b7c8a263b2c",
"key_name": "KeyPair-d750",
"os-extended-volumes:volumes_attached": [{
  "device": "/dev/vda",
  "bootIndex": "0",
  "id": "80c15cff-2473-4982-a816-d760cad6c42c",
  "delete_on_termination": "false"
}],
"OS-EXT-STS:task_state": null,
"OS-EXT-STS:power_state": 1,
"OS-EXT-STS:vm_state": "active",
"OS-EXT-SRV-ATTR:host": "az21.dc1",
"OS-EXT-SRV-ATTR:instance_name": "instance-0015147f",
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova003@74",
"OS-EXT-SRV-ATTR:user_data": null,
"OS-DCF:diskConfig": "MANUAL",
"OS-EXT-AZ:availability_zone": "az1-dc1", //AZ name
"os:scheduler_hints": {
},
"OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
"OS-EXT-SRV-ATTR:ramdisk_id": "",

"OS-SRV-USG:launched_at": "2018-09-13T07:04:09.197749",
"OS-EXT-SRV-ATTR:kernel_id": "",
"OS-EXT-SRV-ATTR:launch_index": 0,
"host_status": "UP",
"OS-EXT-SRV-ATTR:reservation_id": "r-nrd8b5c4",
"OS-EXT-SRV-ATTR:hostname": "ecs-2ecf",
"sys_tags": [{
  "key": "_sys_enterprise_project_id",
  "value": "0"
}],
"security_groups": [{
  "name": "sg-95ec",
  "id": "6505b5d1-7837-41eb-8a1c-869d4355baa3"
}]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.1.5 Querying Details About ECSs

Function

This API is used to query ECSs according to search criteria and details about the ECSs.

The information can be queried includes ECS billing modes and ECS frozen statuses.

URI

```
GET /v1/{project_id}/cloudservers/detail?  
flavor={flavor}&name={name}&status={status}&limit={limit}&offset={offset}&not-  
tags={not-tags}&reservation_id={reservation_id}&&tags={tags}&ip={ip}
```

[Table 4-19](#) describes the parameters in the URI.

Table 4-19 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Table 4-20 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Specifies a page number. The default value is 1 . The value must be greater than or equal to 0. If the value is 0, the first page is displayed.
flavor	No	String	Specifies the flavor ID. For details about the published flavors, see ECS Types in <i>Elastic Cloud Server User Guide</i> .
name	No	String	Specifies the ECS name, which is fuzzy matched.

Parameter	Mandatory	Type	Description
status	No	String	Specifies the ECS status. Options: ACTIVE, BUILD, ERROR, HARD_REBOOT, MIGRATING, REBOOT, REBUILD, RESIZE, REVERT_RESIZE, SHUTOFF, and VERIFY_RESIZE For details, see A.1 ECS Statuses . NOTE When an ECS is in an intermediate state, the statuses that can be obtained are as follows: <ul style="list-style-type: none">• ACTIVE: ACTIVE, REBOOT, HARD_REBOOT, REBUILD, or MIGRATING• SHUTOFF: SHUTOFF, RESIZE, or REBUILD• ERROR: ERROR or REBUILD• VERIFY_RESIZE: VERIFY_RESIZE or REVERT_RESIZE
limit	No	Integer	Specifies the maximum number of ECSs on one page. Each page contains 25 ECSs by default, and a maximum of 1000 ECSs are returned. For large volumes of data, you are advised to set the value to 100 .
tags	No	String	Obtains the ECSs with specified tags.
not-tags	No	String	Queries ECSs whose tag field does not contain the specified value.
reservation_id	No	String	Specifies the ID returned when ECSs are created in a batch by using OpenStack Nova API. This parameter is used to query ECSs created in a batch.
ip	No	String	Indicates the filtering result for IPv4 addresses, which are fuzzy matched. These IP addresses are private IP addresses of the ECS.

Request

None

Response

[Table 4-21](#) describes the response parameters.

Table 4-21 Response parameters

Parameter	Type	Description
servers	Array of objects	Specifies details about ECSs. For details, see Table 4-18 .
count	Integer	Specifies the total number of ECSs.

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/detail?offset=1&limit=10
```

Example Response

```
{
  "count": 4,
  "servers": [{
    "fault": null,
    "id": "b37fd80e-ac67-4d02-b9f1-9891c9c0fabf",
    "name": "ecs-yuankai2",
    "addresses": {
      "164489f6-cbf7-45b4-b6d0-d407c48cf7fc": [{
        "version": "4",
        "addr": "192.168.0.206",
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:95:88:3f",
        "OS-EXT-IPS:port_id": "7b5d615c-186d-4646-9cb8-444adffe9b92",
        "OS-EXT-IPS:type": "fixed"
      }],
      {
        "version": "4",
        "addr": "192.168.0.8",
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:1d:88:43",
        "OS-EXT-IPS:port_id": "dda2027b-2f03-497b-8d42-620da2baacc3",
        "OS-EXT-IPS:type": "fixed"
      }
    ]
  },
  "flavor": {
    "disk": "0",
    "vcpus": "1",
    "ram": "1024",
    "id": "c1.medium",
    "name": "c1.medium"
  },
  "accessIPv4": "",
  "accessIPv6": "",
  "status": "SHUTOFF",
  "image": {
    "id": "1ce5800a-e487-4c1b-b264-3353a39e2b4b"
  },
  "hostId": "f92345b97fd291f67a29ed735a82a8983f370175d2ba3d18d66893f4",
  "updated": "2018-08-14T07:26:49Z",
  "created": "2018-08-13T13:46:09Z",
  "metadata": {
    "metering.image_id": "af60e0d5-6952-4f3d-b0ed-31bb19d4a692",
    "metering.resourcespeccode": "c1.medium.linux",
    "image_name": "HEC_Public_Cloudinit_CentOS_7.4_64bit",
    "metering.product_id": "00301-253164-0--0",
    "os_bit": "64",
    "lockSourceId": "",
    "lockScene": "",
    "metering.order_id": "CS1808132145NRVRE",
    "lockCheckEndpoint": "",
    "metering.imagetype": "gold",
    "lockSource": ""
  }
}
```

```
"metering.resourcetype": "1",
"vpc_id": "164489f6-cbf7-45b4-b6d0-d407c48cf7fc",
"os_type": "Linux",
"charging_mode": "1"
},
"tags": [],
"description": "ecs-4cff",
"locked": false,
"config_drive": "",
"tenant_id": "edcb94a885a84ed3a3fdf8ea4d2741da",
"user_id": "bb7f23e27e7e46f3aaceb5f53a158bdc",
"os-extended-volumes:volumes_attached": [{
  "device": "/dev/sda",
  "bootIndex": "0",
  "id": "2edc879f-022e-4bd6-b079-95a27564d449",
  "delete_on_termination": "false"
}],
  "OS-EXT-STS:task_state": null,
"OS-EXT-STS:power_state": 4,
"OS-EXT-STS:vm_state": "stopped",
"OS-EXT-SRV-ATTR:host": "az1.dc1",
"OS-EXT-SRV-ATTR:instance_name": "instance-00137941",
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova001@248",
"OS-DCF:diskConfig": "MANUAL",
"OS-EXT-AZ:availability_zone": "az1-dc1", //AZ name
"os:scheduler_hints": {

},
"OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
"OS-EXT-SRV-ATTR:ramdisk_id": "",

"OS-EXT-SRV-ATTR:user_data":
"lyEvYmluL2Jhc2gKZWNObyAncm9vdDokNlRkQ2FzUWQkbm5wVmhhJUFZlNVMwczpXbnJGbnZVZ1FCWk4xTE
o5Vy8wd09WTmFZaWpBRXdtRnhuQmZaTlVZXhBwktVWFVTeVhEeERuSUMzV2JjZEJyQUVBZkZvLy8nlHwgY2
hwYXNzd2QgLUU7",
"OS-SRV-USG:launched_at": "2018-08-13T13:46:46.000000",
"OS-EXT-SRV-ATTR:kernel_id": "",
"OS-EXT-SRV-ATTR:launch_index": 0,
"host_status": "UP",
"OS-EXT-SRV-ATTR:reservation_id": "r-a8mg9vwr",
"OS-EXT-SRV-ATTR:hostname": "ecs-4cff",
"sys_tags": [{
  "key": "_sys_enterprise_project_id",
  "value": "441d5677-b76a-4dd4-a97a-ef7fd633c095"
}],
"security_groups": [{
  "id": "71846bf6-1cda-4515-8590-3707be295e76",
  "name": "Sys-FullAccess"
},
{
  "id": "b1786350-da65-11e7-b312-0255ac101b03",
  "name": "default"
}],
{
  "fault": null,
"id": "8380dcc9-0eac-4407-9f9e-df8c9eddeacd",
"name": "ecs-f680",
"addresses": {
  "164489f6-cbf7-45b4-b6d0-d407c48cf7fc": [{
    "version": "4",
    "addr": "192.168.0.218",
    "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:bb:b3:fe",
    "OS-EXT-IPS:port_id": "240c696f-68d8-4f3f-941d-fecf2b375132",
    "OS-EXT-IPS:type": "fixed"
  ]
}
},
"flavor": {
  "disk": "0",
```

```
"vcpus": "1",
"ram": "1024",
"id": "c1.medium",
"name": "c1.medium"
},
"accessIPv4": "",
"accessIPv6": "",
"status": "SHUTOFF",
"image": {
  "id": "1ce5800a-e487-4c1b-b264-3353a39e2b4b"
},
"hostId": "f92345b97fd291f67a29ed735a82a8983f370175d2ba3d18d66893f4",
"updated": "2018-08-14T03:01:00Z",
"created": "2018-08-13T13:38:29Z",
"metadata": {
  "metering.image_id": "af60e0d5-6952-4f3d-b0ed-31bb19d4a692",
  "metering.imagetype": "gold",
  "metering.resourcespeccode": "c1.medium.linux",
  "image_name": "HEC_Public_Cloudinit_CentOS_7.4_64bit",
  "metering.resourcetype": "1",
  "os_bit": "64",
  "vpc_id": "164489f6-cbf7-45b4-b6d0-d407c48cf7fc",
  "os_type": "Linux",
  "charging_mode": "0"
},
"tags": [],
"description": "ecs-f680",
"locked": false,
"config_drive": "",
"tenant_id": "edcb94a885a84ed3a3fdf8ea4d2741da",
"user_id": "61ee747d36bf421fa25c51a3b9565046",
"os-extended-volumes:volumes_attached": [{
  "device": "/dev/sda",
  "bootIndex": "0",
  "id": "3721b948-9c2f-4980-90ad-b2a16811f58c",
  "delete_on_termination": "false"
}],
  "OS-EXT-STS:task_state": null,
  "OS-EXT-STS:power_state": 4,
  "OS-EXT-STS:vm_state": "stopped",
  "OS-EXT-SRV-ATTR:host": "az1.dc1",
  "OS-EXT-SRV-ATTR:instance_name": "instance-00137937",
  "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova001@248",
  "OS-DCF:diskConfig": "MANUAL",
  "OS-EXT-AZ:availability_zone": "az1-dc1", //AZ name
  "os:scheduler_hints": {
  },
  "OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
  "OS-EXT-SRV-ATTR:ramdisk_id": "",
  "OS-EXT-SRV-ATTR:user_data":
  "lyEvYmluL2Jhc2gKZWNobyAncm9vdDokNiR5aG9aeFikVE00OWlwSGQ2OEFWcjltMTFXNEZrZmFYTENVbEkev
  d0xVTmdSVjhOb0dCem5WOWFsU1EN0ZNSHc0VmtwdU9GOERyLncudGUzVmRHLnVmY005eIVZSDEnIHwgY
  2hwYXNzd2QgLUWU7",
  "OS-SRV-USG:launched_at": "2018-08-13T13:38:53.000000",
  "OS-EXT-SRV-ATTR:kernel_id": "",
  "OS-EXT-SRV-ATTR:launch_index": 0,
  "host_status": "UP",
  "OS-EXT-SRV-ATTR:reservation_id": "r-7e2g78rq",
  "OS-EXT-SRV-ATTR:hostname": "ecs-f680",
  "sys_tags": [{
    "key": "_sys_enterprise_project_id",
    "value": "441d5677-b76a-4dd4-a97a-ef7fd633c095"
  }],
  "security_groups": [{
    "name": "test"
  }
]}
{
```



```
    "fault": null,
    "id": "fb70fed9-5774-44a7-ad4a-af3ea2c2da61",
    "name": "ecs-3993",
    "addresses": {
      "00159d7d-b3c3-4108-8bc4-6658814e6422": [{
        "version": "4",
        "addr": "192.168.20.83",
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:a9:8d:88",
        "OS-EXT-IPS:port_id": "579ab762-bf89-435e-80ad-a8bdd25119c5",
        "OS-EXT-IPS:type": "fixed"
      }]
    },
    "flavor": {
      "disk": "0",
      "vcpus": "1",
      "ram": "1024",
      "id": "c1.medium",
      "name": "c1.medium"
    },
    "accessIPv4": "",
    "accessIPv6": "",
    "status": "SHUTOFF",
    "image": {
      "id": "1ce5800a-e487-4c1b-b264-3353a39e2b4b"
    },
    "hostId": "f92345b97fd291f67a29ed735a82a8983f370175d2ba3d18d66893f4",
    "updated": "2018-08-14T03:01:03Z",
    "created": "2018-08-13T13:38:02Z",
    "metadata": {
      "metering.image_id": "af60e0d5-6952-4f3d-b0ed-31bb19d4a692",
      "metering.imagetype": "gold",
      "metering.resourcespeccode": "c1.medium.linux",
      "image_name": "HEC_Public_Cloudinit_CentOS_7.4_64bit",
      "metering.resourcetype": "1",
      "os_bit": "64",
      "vpc_id": "00159d7d-b3c3-4108-8bc4-6658814e6422",
      "os_type": "Linux",
      "charging_mode": "0"
    },
    "tags": [],
    "description": "ecs-3993",
    "locked": false,
    "config_drive": "",
    "tenant_id": "edcb94a885a84ed3a3fdf8ea4d2741da",
    "user_id": "eb4698fe015848e9a3e86cc9956e54fa",
    "key_name": "KeyPair-3b38",
    "os-extended-volumes:volumes_attached": [{
      "device": "/dev/sda",
      "bootIndex": "0",
      "id": "85bfb4c4f-7733-419a-b171-c00585abf926",
      "delete_on_termination": "false"
    }],
    "OS-EXT-STS:task_state": null,
    "OS-EXT-STS:power_state": 4,
    "OS-EXT-STS:vm_state": "stopped",
    "OS-EXT-SRV-ATTR:host": "az1.dc1",
    "OS-EXT-SRV-ATTR:instance_name": "instance-00137936",
    "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova001@248",
    "OS-DCF:diskConfig": "MANUAL",
    "OS-EXT-AZ:availability_zone": "az1-dc1", //AZ name
    "os:scheduler_hints": {
    },
    "OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
    "OS-EXT-SRV-ATTR:ramdisk_id": "",

    "OS-SRV-USG:launched_at": "2018-08-13T13:38:24.000000",
    "OS-EXT-SRV-ATTR:kernel_id": "",
    "OS-EXT-SRV-ATTR:launch_index": 0,
    "host_status": "UP",
```

```
"OS-EXT-SRV-ATTR:reservation_id": "r-uzsewxii",
"OS-EXT-SRV-ATTR:hostname": "ecs-3993",
"sys_tags": [{
  "key": "_sys_enterprise_project_id",
  "value": "441d5677-b76a-4dd4-a97a-ef7fd633c095"
}],
"security_groups": [{
  "name": "test"
},
{
  "name": "default"
}]
},
{
  "fault": null,
"id": "e3d3f219-b445-4a7a-8f00-e31412481f8c",
"name": "ecs-1f30",
"addresses": {
  "00159d7d-b3c3-4108-8bc4-6658814e6422": [{
    "version": "4",
    "addr": "192.168.20.197",
    "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:41:5a:32",
    "OS-EXT-IPS:port_id": "cfa2e055-54fb-427a-bde4-128bda47ae5c",
    "OS-EXT-IPS:type": "fixed"
  ]
},
"flavor": {
  "disk": "0",
  "vcpus": "1",
  "ram": "1024",
  "id": "c1.medium",
  "name": "c1.medium"
},
"accessIPv4": "",
"accessIPv6": "",
"status": "ACTIVE",
  "image": {
    "id": "1ce5800a-e487-4c1b-b264-3353a39e2b4b"
  },
"progress": 0,
"hostId": "f92345b97fd291f67a29ed735a82a8983f370175d2ba3d18d66893f4",
"updated": "2018-08-15T08:16:01Z",
"created": "2018-08-13T11:57:29Z",
"metadata": {
  "sdfasf": "sdffffd",
  "metering.order_id": "CS180813193577ORO",
  "metering.imagetype": "gold",
  "metering.resourcespeccode": "c1.medium.win",
  "metering.image_id": "65cb40e6-f67e-4bef-a1e7-808166a5999d",
  "image_name": "HEC_Public_Windows2008R2_Ent_64bit40G_English",
  "aaaaa": "0",
  "metering.resourcetype": "1",
  "aaaa": "0",
  "metering.product_id": "00301-146042-0--0",
  "os_bit": "64",
  "vpc_id": "00159d7d-b3c3-4108-8bc4-6658814e6422",
  "os_type": "Windows",
  "charging_mode": "1"
},
"tags": [],
"description": "ecs-1f30",
"locked": false,
"config_drive": "",
"tenant_id": "edcb94a885a84ed3a3fdf8ea4d2741da",
"user_id": "bb7f23e27e7e46f3aaceb5f53a158bdc",
"key_name": "Autotest_Init_TC_OriginalAPI_Create_Keypairs_02_keypair",
"os-extended-volumes:volumes_attached": [{
  "device": "/dev/sda",
  "bootIndex": "0",
```

```
    "id": "5043f66b-a0d8-4eb2-8c48-49976bcd253",
    "delete_on_termination": "false"
  }],
  "OS-EXT-STS:task_state": null,
  "OS-EXT-STS:power_state": 1,
  "OS-EXT-STS:vm_state": "active",
  "OS-EXT-SRV-ATTR:host": "az1.dc1",
  "OS-EXT-SRV-ATTR:instance_name": "instance-0013772d",
  "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova001@248",
  "OS-DCF:diskConfig": "MANUAL",
  "OS-EXT-AZ:availability_zone": "az1-dc1", //AZ name
  "os:scheduler_hints": {
  },
  "OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
  "OS-EXT-SRV-ATTR:ramdisk_id": "",

  "OS-SRV-USG:launched_at": "2018-08-13T11:57:53.576640",
  "OS-EXT-SRV-ATTR:kernel_id": "",
  "OS-EXT-SRV-ATTR:launch_index": 0,
  "host_status": "UP",
  "OS-EXT-SRV-ATTR:reservation_id": "r-xmjj4pnm",
  "OS-EXT-SRV-ATTR:hostname": "ecs-1f30",
  "sys_tags": [{
    "key": "_sys_enterprise_project_id",
    "value": "441d5677-b76a-4dd4-a97a-ef7fd633c095"
  }],
  "security_groups": [{
    "name": "default"
  }]
}]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.1.6 Modifying an ECS

Function

This API is used to modify ECS information. Only the name, description, and hostname of an ECS can be modified currently.

Constraints

After the hostname of an ECS is changed, you need to restart the ECS for the configuration to take effect.

URI

PUT /v1/{project_id}/cloudservers/{server_id}

[Table 4-22](#) describes the parameters in the URI.

Table 4-22 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-23](#) describes the request parameters.

Table 4-23 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS data structure. For details, see Table 4-24 .

Table 4-24 server field description

Parameter	Mandatory	Type	Description
name	No	String	Specifies the name of the modified ECS. The parameter value consists of 1 to 64 characters, including letters, digits, underscores (_), hyphens (-), periods (.).
description	No	String	Describes the ECS. The value consists of 0-85 characters and cannot contain brackets (<>).
hostname	No	String	Specifies the ECS hostname. The name consists of 1-64 characters. It can be segmented using periods (.). Only letters, digits, and hyphens (-) are allowed in each segment. A name cannot contain consecutive periods (.) or hyphens (-), and cannot start or end with a period (.) or hyphen (-). Additionally, the combinations of (-.) and (-.) are not allowed.

Response

[Table 4-25](#) describes the response parameters.

Table 4-25 Response parameters

Parameter	Type	Description
server	Object	Specifies the ECS. For details, see Table 4-26 .

Table 4-26 server field description

Parameter	Type	Description
tenant_id	String	Specifies the tenant or project ID.
image	String	Specifies the image ID.
accessIPv4	String	Reserved
addresses	Object	Specifies the network addresses of an ECS. For details, see Table 4-27 .
metadata	Object	Specifies the ECS metadata.
accessIPv6	String	Reserved
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 4-29 .
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 name of the modified ECS.
progress	Integer	Reserved
links	Array of Object	Specifies ECS shortcut links for ECS. For details, see Table 4-30 .
id	String	Specifies the unique ID of an ECS.
updated	String	Specifies the time when the ECS was updated last time. The time is in the format of "2019-05-22T03:19:19Z".

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 field is supported in microversions later than 2.9.
description	String	Describes the ECS. This field is supported in microversions later than 2.19.
tags	Array of strings	Specifies ECS tags. This field is supported in microversions later than 2.26. If the microversion is not used for query, the response does not contain the tags field. Tag functions have been upgraded on the public cloud. After the upgrade, the tag values returned by the system comply with the following rules: <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. • 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 A.1 ECS Statuses .
OS-EXT-SRV-ATTR:hostname	String	Specifies the ECS hostname.

Table 4-27 addresses field description

Parameter	Type	Description
Name of the network where the ECS accesses	Object	Specifies the network information of the ECS. <ul style="list-style-type: none">• key indicates the network name, for example, demo_net.• value indicates the detailed network information. For details, see Table 4-28 .

Table 4-28 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 4-29 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 4-30 .

Table 4-30 links field description

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

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/{server_id}
{
  "server": {
    "name": "new-server-test"
  }
}
```

Example Response

```
{
  "server": {
    "tenant_id": "66c860cb130b465fbafcdee43fb09c64",
    "image": "",
    "accessIPv4": "",
    "addresses": {
      "01d7aef8-442b-408e-b82f-13afff51e4e4": [
        {
          "addr": "192.168.26.22",
          "version": 4
        }
      ]
    },
    "metadata": {
      "virtual_env_type": "FusionCompute"
    },
    "description": "",
    "accessIPv6": "",
    "created": "2019-04-25T11:52:53Z",
    "hostId": "57d278e7c53d07cd34fad3ba4fdc9f3d779017d0879726d83b45a22a",
    "OS-EXT-SRV-ATTR:hostname": "new-test-hostname",
    "flavor": {
      "links": [
        {
          "rel": "bookmark",
          "href": "https://None/66c860cb130b465fbafcdee43fb09c64/flavors/s2.large.2"
        }
      ],
      "id": "s2.large.2"
    },
    "OS-DCF:diskConfig": "MANUAL",
    "user_id": "f88581d53be64716a985c66ca28c75f6",
    "name": "new-test-hostname",
    "progress": 0,
    "links": [
      {
        "rel": "self",
        "href": "https://None/v2/66c860cb130b465fbafcdee43fb09c64/servers/24930df0-db4c-4a8b-8914-d0bd558564b0"
      },
      {
        "rel": "bookmark",
        "href": "https://None/66c860cb130b465fbafcdee43fb09c64/servers/24930df0-db4c-4a8b-8914-d0bd558564b0"
      }
    ],
    "id": "24930df0-db4c-4a8b-8914-d0bd558564b0",
    "updated": "2019-04-28T08:15:36Z",
    "status": "ACTIVE"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2 Status Management

4.2.1 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.

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. If the image has no Cloudbase-Init or Cloudbase-init installed, use the API described in [4.2.3 Reinstalling an ECS OS \(Using an Image Without Cloud-Init Installed\)](#).
- 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 4-31](#) describes the parameters in the URI.

Table 4-31 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-32](#) describes the request parameters.

Table 4-32 Request parameters

Parameter	Mandatory	Type	Description
os-reinstall	Yes	Object	Reinstalls an ECS OS. For details, see Table 4-33 .

Table 4-33 os-reinstall field description

Parameter	Mandatory	Type	Description
adminpass	No	String	<p>Specifies the initial password of the ECS administrator.</p> <p>The Windows administrator username is Administrator, and the Linux administrator username is root.</p> <p>Password complexity requirements:</p> <ul style="list-style-type: none"> • 8 to 26 characters • The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@\$%^_-=+[]{};,:./?~#*). <p>NOTE</p> <ul style="list-style-type: none"> • The Windows ECS password cannot contain the username, the username in reverse, or more than two characters in the same sequence as they appear in the username. • user_data can be used by Linux ECSs to inject passwords. In such a case, adminpass is unavailable. • Either adminpass or keyname is set. • If both adminpass and keyname are empty, user_data in metadata must be set.
keyname	No	String	<p>Specifies the key pair name.</p> <p>Keys can be created using the key creating API (5.11.3 Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (5.11.1 Querying SSH Key Pairs).</p>
userid	No	String	Specifies the user ID.
metadata	No	Object	Specifies metadata of the reinstalled ECS. For more information, see Table 4-34 .

Parameter	Mandatory	Type	Description
mode	No	String	Specifies whether the ECS supports OS reinstallation when the ECS is running. If the parameter value is withStopServer , the ECS supports OS reinstallation when the ECS is running. In such a case, the system automatically stops the ECS before reinstalling its OS.

Table 4-34 metadata field description

Parameter	Mandatory	Type	Description
user_data	No	String	Specifies the user data to be injected during the ECS creation. Text and text files can be injected. NOTE <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. For more details, see Injecting User Data into ECSs . Examples Before base64 encoding: <ul style="list-style-type: none"> Linux #!/bin/bash echo user_test >> /home/user.txt Windows rem cmd echo 111 > c:\aaa.txt After base64 encoding: <ul style="list-style-type: none"> Linux lyEgL2Jpbi9iYXNoDQpLY2hvlHVzZXJfdGVzd-CAmZ3Q7Jmd0OyAvaG9tZS91c2VyLnR4dA== Windows cmVtIGNtZA0KZWNoYAxMTEgJmd0OyBjOlxhYWEudHh0

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

- Example URL request
POST `https://{endpoint}/v2/{project_id}/cloudservers/{server_id}/reinstallos`
- Example request 1 (using a password to remotely log in to an ECS with OS reinstalled)

```
{
  "os-reinstall": {
    "adminpass": "!QAZxsw2",
    "userid": "7e25b1da389f4697a79df3a0e5bd494e",
    "mode": "withStopServer"
  }
}
```

- Example request 2 (using a key to remotely log in to an ECS with OS reinstalled)

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

Example Response

See [9.5.1 Responses \(Task\)](#).

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.2 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.

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. If the image has no Cloudbase-Init or Cloudbase-init installed, use the API described in [4.2.4 Changing an ECS OS \(Using an Image Without Cloud-Init Installed\)](#).
- 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 4-35](#) describes the parameters in the URI.

Table 4-35 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-36](#) describes the request parameters.

Table 4-36 Request parameters

Parameter	Mandatory	Type	Description
os-change	Yes	Object	Changes an ECS OS. For details, see Table 4-37 .

Table 4-37 os-change field description

Parameter	Mandatory	Type	Description
adminpass	No	String	<p>Specifies the initial password of the ECS administrator.</p> <p>The Windows administrator username is Administrator, and the Linux administrator username is root.</p> <p>The password must meet the following requirements:</p> <ul style="list-style-type: none"> • 8 to 26 characters • The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@\$%^_+=+[{ }];,./?~#*). <p>NOTE</p> <ul style="list-style-type: none"> • The Windows ECS password cannot contain the username, the username in reverse, or more than two characters in the same sequence as they appear in the username. • user_data can be used by Linux ECSs to inject passwords. In such a case, adminpass is unavailable. • Either adminpass or keyname is set. • If both adminpass and keyname are empty, user_data in metadata must be set. • If you use this field to change the OS of an ECS with Cloud-Init installed, the region in which the ECS is deployed does not support password-authenticated OS changing. In such a case, use key pair authentication.
keyname	No	String	<p>Specifies the key pair name.</p> <p>Keys can be created using the key creating API (5.11.3 Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (5.11.1 Querying SSH Key Pairs).</p>
userid	No	String	<p>Specifies the user ID. This parameter is mandatory when keyname is used.</p>
imageid	Yes	String	<p>Specifies the ID of the new image in UUID format.</p> <p>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>.</p>

Parameter	Mandatory	Type	Description
metadata	No	Object	Specifies the metadata of the ECS for which the OS is to be changed. For more information, see Table 4-38 .
mode	No	String	Specifies whether the ECS supports OS change when the ECS is running. If the parameter value is withStopServer , the ECS supports this feature. The system automatically stops the ECS and then change its OS.

Table 4-38 metadata field description

Parameter	Mandatory	Type	Description
user_data	No	String	<p>Specifies the user data to be injected during the ECS 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.</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 lyEgL2Jpbi9iYXNoDQplY2hvlHVzZXJfdGVzd-CAmZ3Q7Jmd0OyAvaG9tZS91c2VyLnR4dA== Windows cmVtIGNtZA0KZWNoYAxMTEgJmd0OyBjOlxhYW EudHh0

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

- Example URL request
POST `https://{endpoint}/v2/{project_id}/cloudservers/{server_id}/changeos`
- Example request 1 (using a password to remotely log in to an ECS with OS changed)

```
{
  "os-change": {
    "adminpass": "1qazXSW@",
    "userid": "7e25b1da389f4697a79df3a0e5bd494e",
    "imageid": "e215580f-73ad-429d-b6f2-5433947433b0",
    "mode": "withStopServer"
  }
}
```

- Example request 2 (using a key to remotely log in to an ECS with OS changed)

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

Example Response

See [9.5.1 Responses \(Task\)](#).

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.3 Reinstalling an ECS OS (Using an Image Without Cloud-Init Installed)

Function

This API is used to reinstall an ECS OS.

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.

This API supports the images with Cloud-Init or Cloudbase-Init installed. Otherwise, use the API described in [4.2.1 Reinstalling an ECS OS \(Using an Image with Cloud-Init Installed\)](#).

Constraints

- You cannot reinstall OS on 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 /v1/{project_id}/cloudservers/{server_id}/reinstall

[Table 4-39](#) describes the parameters in the URI.

Table 4-39 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-40](#) describes the request parameters.

Table 4-40 Request parameters

Parameter	Mandatory	Type	Description
os-reinstall	Yes	Object	Reinstall the ECS. For details, see Table 4-41 .

Table 4-41 os-reinstall field description

Parameter	Mandatory	Type	Description
adminpass	No	String	<p>Specifies the initial password of the ECS administrator.</p> <p>The Windows administrator username is Administrator, and the Linux administrator username is root.</p> <p>Password complexity requirements:</p> <ul style="list-style-type: none">• 8 to 26 characters• The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@\$%^&_+=+[{ }],./?~#*). <p>NOTE</p> <ul style="list-style-type: none">• You can only log in to a Windows ECS using a username and password, and the password cannot contain the username, the username in reverse, or more than two characters in the same sequence as they appear in the username.• Either adminpass or keyname is empty.• Either adminpass or keyname is set.
keyname	No	String	<p>Specifies the key name.</p> <p>Keys can be created using the key creating API (5.11.3 Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (5.11.1 Querying SSH Key Pairs).</p>
userid	No	String	Specifies the user ID.
mode	No	String	<p>Specifies whether the ECS supports OS reinstallation when the ECS is running.</p> <p>If the parameter value is withStopServer, the ECS supports OS reinstallation when the ECS is running. In such a case, the system automatically stops the ECS before reinstalling its OS.</p>

Response

For details, see [9.5.1 Responses \(Task\)](#).

Example Request

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/reinstallos
{
  "os-reinstall": {
```

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

Example Response

See [9.5.1 Responses \(Task\)](#).

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.4 Changing an ECS OS (Using an Image Without Cloud-Init Installed)

Function

This API is used to change the OS of an ECS.

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.

This API supports the images with Cloud-Init or Cloudbase-Init installed. Otherwise, use the API described in [4.2.2 Changing an ECS OS \(Using an Image with Cloud-Init Installed\)](#).

Constraints

- 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 /v1/{project_id}/cloudservers/{server_id}/changeos

[Table 4-42](#) describes the parameters in the URI.

Table 4-42 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-43](#) describes the request parameters.

Table 4-43 Request parameters

Parameter	Mandatory	Type	Description
os-change	Yes	Object	Changes the OS of an ECS. For details, see Table 4-44 .

Table 4-44 os-change field description

Parameter	Mandatory	Type	Description
adminpass	No	String	<p>Specifies the initial password of the ECS administrator.</p> <p>The Windows administrator username is Administrator, and the Linux administrator username is root.</p> <p>Password complexity requirements:</p> <ul style="list-style-type: none">• 8 to 26 characters• The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@\$%^-_=+[{ }];,./?~#*). <p>NOTE</p> <ul style="list-style-type: none">• You can only log in to a Windows ECS using a username and password, and the password cannot contain the username, the username in reverse, or more than two characters in the same sequence as they appear in the username.• Either adminpass or keyname is empty.• Either adminpass or keyname is set.
keyname	No	String	<p>Specifies the key name.</p> <p>Keys can be created using the key creating API (5.11.3 Creating and Importing an SSH Key Pair) or obtained using the SSH key query API (5.11.1 Querying SSH Key Pairs).</p>
userid	No	String	<p>Specifies the user ID. This parameter is mandatory when keyname is used.</p>

Parameter	Mandatory	Type	Description
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> .
mode	No	String	Specifies whether the ECS supports OS change when the ECS is running. If the parameter value is withStopServer , the ECS supports OS change when the ECS is running. In such a case, the system automatically stops the ECS before changing its OS.

Response

For details, see [9.5.1 Responses \(Task\)](#).

Example Request

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/changeos
{
  "os-change": {
    "keyname": "KeyPair-350b",
    "userid": "7e25b1da389f4697a79df3a0e5bd494e",
    "imageid": "e215580f-73ad-429d-b6f2-5433947433b0"
  }
}
```

Example Response

See [9.5.1 Responses \(Task\)](#).

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.5 Querying Automatic Recovery of an ECS

Function

This API is used to query automatic recovery configured for an ECS.

URI

GET /v1/{project_id}/cloudservers/{server_id}/autorecovery

[Table 4-45](#) describes the parameters in the URI.

Table 4-45 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-46](#) describes the response parameters.

Table 4-46 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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.6 Managing Automatic Recovery of an ECS

Function

This API is used to configure or delete automatic recovery of an ECS.

URI

PUT /v1/{project_id}/cloudservers/{server_id}/autorecovery

[Table 4-47](#) describes the parameters in the URI.

Table 4-47 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-48](#) describes the request parameters.

Table 4-48 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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.7 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.

Constraints

- This API is supported by DeHs only.
- 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 4-49](#) describes the parameters in the URI.

Table 4-49 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-50](#) describes the request parameters.

Table 4-50 Request parameters

Parameter	Mandatory	Type	Description
migrate	Yes	Object	Specifies the ECS to be migrated. For details, see Table 4-51 . When migrating an ECS from a DeH to a public resource pool, the migrate value is null.

Table 4-51 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 when an ECS is migrated between DeHs.

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.8 Obtaining the VNC Login Address

Function

This API is used to obtain the address for remotely logging in to an ECS using VNC.

URI

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

[Table 4-52](#) describes the parameters in the URI.

Table 4-52 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

Request parameters

[Table 4-53](#) describes the request parameters.

Table 4-53 Request parameters

Parameter	Mandatory	Type	Description
remote_console	Yes	Object	Obtains the address for remotely logging in to an ECS. For details, see Table 4-54 .

Table 4-54 remote_console parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies a remote login mode. Set it to novnc .
protocol	Yes	String	Specifies a remote login protocol. Set it to vnc .

Response

Response parameters

[Table 4-55](#) describes the response parameters.

Table 4-55 Response parameters

Parameter	Type	Description
remote_console	Object	Obtains the address for remotely logging in to an ECS. For details, see Table 4-56 .

Table 4-56 remote_console field description

Parameter	Type	Description
type	String	Specifies a remote login mode.
protocol	String	Specifies a remote login protocol.
url	String	Specifies a remote login URL.

Example Request

```
POST https://{endpoint}/v1/13c67a214ced4afb88d911ae4bd5721a/cloudservers/47bc79ae-  
df61-4ade-9197-283a74e5d70e/remote_console  
{  
  "remote_console": {  
    "protocol": "vnc",  
    "type": "novnc"  
  }  
}
```

Example Response

```
{  
  "remote_console": {  
    "type": "novnc",  
    "protocol": "vnc",  
    "url": "https://nova-novncproxy.az1.dc1.domainname.com:8002/vnc_auto.html?  
token=0fda3eca-8232-4249-a69f-44ce8ab31be6&lang=EN&tLength=70"  
  }  
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.9 Modifying the Specifications of an ECS

Function

This API is used to modify ECS specifications.

The V1.1 API supports all functions (see [4.2.10 Modifying the Specifications of an ECS \(Pay-per-Use\)](#)) provided by the V1 API. Additionally, the V1.1 API supports the modification of yearly/monthly ECSs.

Constraints

- You can modify the ECS specifications only when the ECS is stopped.
- Spot ECSs do not support specifications modification.

URI

POST /v1.1/{project_id}/cloudservers/{server_id}/resize

Table 4-57 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID. You can obtain the ECS ID from the console or by following the instructions provided in "Querying Details About an ECS".

Request

Parameter description

Parameter	Mandatory	Type	Description
resize	Yes	Object	Specifies the operation to modify ECS specifications. For details, see Table 4-58 .

Table 4-58 resize field description

Parameter	Mandatory	Type	Description
flavorRef	Yes	String	Specifies the flavor ID of the ECS after the modification. You can view 4.4.2 Querying the Target ECS Flavors to Which a Flavor Can Be Changed to query the target flavors to which a specified ECS flavor can be changed. NOTE <ul style="list-style-type: none"> Modifications between the same flavor are not supported.
dedicated_host_id	No	String	Specifies the new DeH ID, which is applicable only to the ECSs on DeHs.
extendparam	No	Object	Modify the extended information about an ECS. For details, see Table 4-59 .

Table 4-59 extendparam field description

Parameter	Mandatory	Type	Description
isAutoPay	No	String	Specifies whether the order is automatically or manually paid. <ul style="list-style-type: none"> true: The order will be automatically paid. false: The order must be manually paid. NOTE This parameter is valid only for yearly/monthly ECSs. When this parameter is left blank, the order must be manually paid by default.

Response

Table 4-60 Parameter description

Parameter	Mandatory	Type	Description
job_id	No	String	Specifies the task ID. This parameter is returned when you modify the specifications of a pay-per-use ECS. For details about task statuses, see 4.9.1 Querying Task Execution Status .

Parameter	Mandatory	Type	Description
order_id	No	String	Specifies the order ID. This parameter is returned when you modify the specifications of a yearly/monthly ECS.

Example Request

```
POST https://{endpoint}/v1.1/{project_id}/cloudservers/{server_id}/resize
{
  "resize": {
    "flavorRef": "s3.large.2",
    "dedicated_host_id": "459a2b9d-804a-4745-ab19-a113bb1b4ddc",
    "extendparam":{
      "isAutoPay": "true"
    },
  },
}
```

Example Response

```
{
  "job_id": "70a599e0-31e7-49b7-b260-868f441e862b"
}
```

Or

```
{
  "order_id": "CS1711152257C60TL",
  "job_id": "70a599e0-31e7-49b7-b260-868f441e862b"
}
```

Or

```
{
  "error": {
    "message": "XXXX",
    "code": "XXX"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.10 Modifying the Specifications of an ECS (Pay-per-Use)

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.

Constraints

- You can modify the ECS specifications only when the ECS is stopped.
- This API cannot be used to modify the specifications of a yearly/monthly ECS. For details about how to modify the specifications of a yearly/monthly ECS, see [4.2.9 Modifying the Specifications of an ECS](#).
- Spot ECSs do not support specifications modification.

URI

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

[Table 4-61](#) describes the parameters in the URI.

Table 4-61 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-62](#) describes the request parameters.

Table 4-62 Request parameters

Parameter	Mandatory	Type	Description
resize	Yes	Object	Specifies the operation to modify ECS specifications. For details, see Table 4-63 .

Table 4-63 resize field description

Parameter	Mandatory	Type	Description
flavorRef	Yes	String	Specifies the flavor ID of the ECS after the modification. You can view 4.4.2 Querying the Target ECS Flavors to Which a Flavor Can Be Changed to query the target flavors to which a specified ECS flavor can be changed.

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.2.11 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 4-64](#) describes the parameters in the URI.

Table 4-64 Parameter description

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-65](#) describes the request parameters.

Table 4-65 Request parameters

Parameter	Mandatory	Type	Description
monitorMetrics	Yes	Null	Enables monitoring on the ECS.

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.3 Batch Operations

4.3.1 Starting ECSs in a Batch

Function

This API is used to start ECSs in a batch based on specified ECS IDs. A maximum of 1000 ECSs can be started at a time.

URI

```
POST /v1/{project_id}/cloudservers/action
```

[Table 4-66](#) describes the parameters in the URI.

Table 4-66 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

Table 4-67 Request parameters

Parameter	Mandatory	Type	Description
os-start	Yes	Object	Specifies the operation to start the ECS. For details, see Table 4-68 .

Table 4-68 os-start field description

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies ECS IDs. For details, see Table 4-69 .

Table 4-69 servers field description

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

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

In the request, the parameters to start ECSs must be sent with field **os-start**. For details, see the example request.

```
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

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.3.2 Restarting ECSs in a Batch

Function

This API is used to restart ECSs in a batch based on specified ECS IDs. A maximum of 1000 ECSs can be restarted at a time.

URI

POST /v1/{project_id}/cloudservers/action

[Table 4-70](#) describes the parameters in the URI.

Table 4-70 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 4-71](#) describes the request parameters.

Table 4-71 Request parameters

Parameter	Mandatory	Type	Description
reboot	Yes	Object	Specifies the operation to restart the ECS. For details, see Table 4-72 .

Table 4-72 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)
servers	Yes	Array of objects	Specifies ECS IDs. For details, see Table 4-73 .

Table 4-73 servers field description

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

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

In the request, the parameters to restart ECSs must be sent with field **reboot**. For details, see the example request.

```
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

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.3.3 Stopping ECSs in a Batch

Function

This API is used to stop ECSs in a batch based on the specified ECS ID list. A maximum of 1000 ECSs can be stopped at a time.

URI

POST /v1/{project_id}/cloudservers/action

[Table 4-74](#) describes the parameters in the URI.

Table 4-74 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

Table 4-75 Request parameters

Parameter	Mandatory	Type	Description
os-stop	Yes	Object	Specifies the operation to stop the ECS. For details, see Table 4-76 .

Table 4-76 os-stop field description

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies ECS IDs. For details, see Table 4-77 .
type	No	String	Specifies an ECS stop type. The default value is SOFT . SOFT : normal ECS stop (default) HARD : forcible ECS stop

Table 4-77 servers field description

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

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

In the request parameters, the request for stopping the ECS must be issued with field **os-stop**, as shown in the example request.

```
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

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.3.4 Modifying ECSs in a Batch

Function

This API is used to modify ECSs in a batch.

Only ECS names can be changed in a batch, and the maximum number is 1000 at a time.

URI

PUT /v1/{project_id}/cloudservers/server-name

[Table 4-78](#) lists the URI parameters.

Table 4-78 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 4-79](#) describes the request parameters.

Table 4-79 Request parameters

Parameter	Type	Mandatory	Description
name	String	Yes	Specifies the changed name of the ECSs. The rules are as follows: Consists of a maximum of 64 characters, including uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.). After you change ECS names in a batch, the system does not automatically add a digital suffix to the changed names. For example, there are three ECSs, test_0001 , test_0002 , and test_0003 . After their names are changed to develop in a batch, their changed names are all develop .
dry_run	Boolean	No	Specifies whether to check the request and change ECS names. true : indicates that only the name change request is sent and the names of the ECSs will not be changed. 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. See 9.6.1 Responses (Batch Operation) . false : indicates that the name change request is sent and the ECS names will be changed if the check result is as expected. The default value is false .

Parameter	Type	Mandatory	Description
servers	Array of objects	Yes	Specifies the IDs of the target ECSs. For details, see Table 4-80 .

Table 4-80 servers field description

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

Response

See [9.6.1 Responses \(Batch Operation\)](#).

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/server-name
{
  "name": "new-server-name",
  "dry_run": false,
  "servers": [
    {
      "id": "260a0917-f7df-4b25-93ac-950da6c6b5d6"
    },
    {
      "id": "f6d8df1a-e257-48e2-b617-1dd92ced8c20"
    }
  ]
}
```

Example Response

See [9.6.1 Responses \(Batch Operation\)](#).

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.3.5 Resetting the Passwords for Logging In to ECSs in a Batch

Function

This API is used to reset the passwords of the ECS management account, **root** or **Administrator**, in a batch.

Constraints

- Before using this API, you must install password reset plug-ins. For instructions about how to download and install the password reset plug-ins, see "Installing One-Click Password Reset Plug-ins" in *Elastic Cloud Server User Guide*.
- After the request for resetting the password is issued, this API does not report an error if executing the script failed.
- A new password takes effect after the ECS is started or restarted.

URI

PUT /v1/{project_id}/cloudservers/os-reset-passwords

[Table 4-81](#) lists the URI parameters.

Table 4-81 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 4-82](#) describes the request parameters.

Table 4-82 Request parameters

Parameter	Type	Mandatory	Description
new_password	String	Yes	<p>Specifies the new password.</p> <p>This field is mandatory only if dry_run is set to false.</p> <p>A new password must comply with the following rules:</p> <ul style="list-style-type: none">• Consists of 8 to 26 characters.• Supports the following characters: ! @%_-_+[]:./?• Cannot contain any of the following characters: Chinese characters and ~`#&^,{}*()'"<> \\$• Contains at least three of the following: uppercase letters, lowercase letters, digits, and allowed special characters.• Cannot contain username Administrator/root or the username spelled backwards.• Cannot contain three consecutive characters in username Administrator
dry_run	Boolean	No	<p>Specifies whether to check the request and reset ECS passwords.</p> <ul style="list-style-type: none">• true: indicates that only the password reset request is sent and the passwords for logging in to the ECSs will not be reset. 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 password reset request is sent and the passwords for logging in to the ECSs will be reset if the check result is as expected. <p>The default value is false.</p>
servers	Array of objects	Yes	<p>Specifies the IDs of the target ECSs. For details, see Table 4-83.</p>

Table 4-83 servers field description

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

Response

See [9.6.1 Responses \(Batch Operation\)](#).

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/os-reset-passwords
{
  "new_password": "Test@123",
  "dry_run": true,
  "servers": [
    {
      "id": "1bd0eb17-4466-4c15-a9ce-87727ad311b5"
    },
    {
      "id": "fd6b6e9d-64a1-40fa-b7dc-f491be42fdd2"
    }
  ]
}
```

Example Response

See [9.6.1 Responses \(Batch Operation\)](#).

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.3.6 Attaching a Specified Shared EVS Disk to Multiple ECSs

Function

This API is used to attach a specified shared EVS disk to multiple ECSs.

Constraints

No more than 23 disks have been attached to each of these ECSs.

URI

POST /v1/{project_id}/batchaction/attachvolumes/{volume_id}

[Table 4-84](#) describes the parameters in the URI.

Table 4-84 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
volume_id	Yes	Specifies the shared EVS disk ID.

Request

[Table 4-85](#) describes the request parameters.

Table 4-85 Request parameters

Parameter	Mandatory	Type	Description
serverinfo	Yes	Array of objects	Specifies the list of ECSs to which the shared EVS disk is to be attached. For details, see Table 4-86 .

Table 4-86 serverinfo field description

Parameter	Mandatory	Type	Description
server_id	Yes	String	Specifies the ID of the ECS to which the shared EVS disk is to be attached.

Parameter	Mandatory	Type	Description
device	Yes	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.

Response

For details, see [9.5.1 Responses \(Task\)](#).

Example Request

```
POST https://{endpoint}/v1/{project_id}/batchaction/attachvolumes/{volume_id}
{
  "serverinfo": [
    {
      "server_id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
      "device": "/dev/sdb"
    },
    {
      "server_id": "a26887c6-c47b-4654-abb5-dfadf7d3fa05",
      "device": "/dev/sdb"
    }
  ]
}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.4 Flavor Management

4.4.1 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 4-87](#) describes the parameters in the URI.

Table 4-87 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Table 4-88 Query parameters

Parameter	Mandatory	Type	Description
availability_zone	No	String	Specifies the AZ name, ID, or code. See Regions and Endpoints .

Request

None

Response

[Table 4-89](#) describes the response parameters.

Table 4-89 Response parameters

Parameter	Type	Description
flavors	Array of objects	Specifies ECS flavors. For details, see Table 4-90 .

Table 4-90 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 .
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 .

Parameter	Type	Description
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 4-91 .
os_extra_specs	Objects	Specifies extended ECS specifications. For details, see Table 4-92 .
attachable Quantity	Objects	Specifies the number of NICs and disks that can be attached to an ECS. For details, see Table 4-93 .

Table 4-91 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 4-92 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• cpu1: computing I• cpu2: computing II• computingv3: general computing-plus• highmem: memory-optimized• saphana: large-memory HANA ECS• diskintensive: disk-intensive
resource_type	String	Specifies the resource type. resource_type is used to differentiate between the types of the physical servers accommodating ECSs.

Parameter	Type	Description
quota:local_disk	String	<p>The value of this parameter is in format of "{type}:{count}:{size}:{safeFormat}", where,</p> <ul style="list-style-type: none">• type specifies the disk type, which can only be HDD.• count specifies the number of local disks. Its value can be 3/6/12/24 for d1 type of disks, 2/4/8/12/16/24 for d2 type of disks, or 2/4/8/12/16/24/28 for d3 type of disks.• size specifies 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 specifies whether the local disks of an ECS have been securely formatted. The value of this parameter can only be FALSE for D1 ECSs or True for D2 and D3 ECSs. <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 specification of the NVME SSD disk, which can be large or small. If the value is large, only I3 ECSs are supported.• num: indicates the number of partitions on the disk.• 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 spec value is small, the value of this parameter is 1/4 or 1/2 of the specification.• safeFormat: indicates whether the local disks of the ECS are securely formatted. If the value is True, only I3 ECSs are supported. <p>NOTE This field is dedicated for ultra-high I/O ECSs.</p>

Parameter	Type	Description
ecs:generation	String	<p>Specifies the generation of an ECS type.</p> <ul style="list-style-type: none"> • s1: general-purpose first-generation • s2: general-purpose second-generation • s3: general-purpose third-generation • s6: general-purpose • sn3: general network-optimized • c3: general computing-plus • c6: general computing-plus • c3ne: general network enhancement • m1: memory-optimized first-generation • m2: memory-optimized second-generation • m3: memory-optimized third-generation • m6: memory-optimized • m3ne: memory network enhancement • h1: high-performance computing first-generation • h2: high-performance computing second-generation • hc2: high-performance computing • h3: high-performance computing • hi3: ultra-high performance computing • d1: disk-intensive first-generation ECSs • d2: disk-intensive second-generation • d3: disk-intensive • kc1: Kunpeng general computing-plus • km1: Kunpeng memory-optimized • g1: GPU-accelerated first-generation • g2: GPU-accelerated second-generation • e3: large-memory • i3: ultra-high I/O <p>NOTE This field is optional.</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>

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 canceled (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 the recommended flavor (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 1000.
quota:min_rate	String	<p>Specified 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 1000.
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 10000, it must be divided by 10000.

Parameter	Type	Description
cond:operation:charge	String	Specifies a billing type. <ul style="list-style-type: none"> Both billing types are supported if this parameter is not set. Yearly/Monthly Pay-per-use
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.

Table 4-93 attachableQuantity field description

Parameter	Type	Description
free_scsi	Integer	Specifies the number of SCSI disks that can be attached.
free_blk	Integer	Specifies the number of VBD disks that can be attached.
free_disk	Integer	Specifies the number of disks that can be attached.
free_nic	Integer	Specifies the number of NICs that can be attached.

 **NOTE**

For ECS specifications, see "ECS Types and Application Scenarios" in *Elastic Cloud Server User Guide*.

Example Request

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

Example Response

```
{
  "flavors": [
    {
      "attachableQuantity": {
        "free_scsi": 60,
        "free_blk": 24,
        "free_disk": 60,
        "free_nic": 12
      },
      "id": "c3.2xlarge.2",
      "name": "c3.2xlarge.2",
      "vcpus": "8",
      "ram": 16384,
      "disk": "0",
      "swap": "",
      "links": [
        {

```

```
        "rel": "self",
        "href": "https://ecs.region.xxx.com/v1.0/743b4c0428d94531b9f2add666642e6b/flavors/
c3.2xlarge.2",
        "type": null
    },
    {
        "rel": "bookmark",
        "href": "https://ecs.region.xxx.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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.4.2 Querying the Target ECS Flavors to Which a Flavor Can Be Changed

Function

An ECS flavor cannot be changed to certain flavors. This API is used to query the target flavors to which a specified ECS flavor can be changed.

URI

```
GET /v1/{project_id}/cloudservers/resize_flavors?
instance_uuid={instance_uuid}&source_flavor_id={source_flavor_id}&source_flavor_
name={source_flavor_name}
```

[Table 4-94](#) describes the parameters in the URI.

Table 4-94 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

 NOTE

One of the **instance_uuid**, **source_flavor_id**, and **source_flavor_name** parameters must be configured. If multiple parameters are configured, the system processes the **instance_uuid**, **source_flavor_id**, and **source_flavor_name** parameters in descending order by default.

If **instance_uuid** is used to query the flavors that can be changed to, only the flavors supporting the image based on which the target ECS is created are returned. The reason is as follows: Images rely on flavors. If an ECS is created using a public image, the API filters the flavors supported by the image.

Table 4-95 describes the query parameters.

Table 4-95 Query parameters

Parameter	Mandatory	Type	Description
instance_uuid	No	String	Specifies the target ECS ID in UUID format.
source_flavor_id	No	String	Specifies the source flavor ID.
source_flavor_name	No	String	Specifies the source flavor name.
sort_key	No	String	Indicates the field for sorting. Options: <ul style="list-style-type: none">● flavorid: indicates the flavor ID. The default value is flavorid.● name: indicates the flavor name.● memory_mb: indicates the memory size.● vcpus: indicates the number of vCPUs.● root_gb: indicates the system disk size.
sort_dir	No	String	Specifies the ascending (asc) or descending (desc) sorting. Options: <ul style="list-style-type: none">● asc: indicates the ascending order.● desc: indicates the descending order.
limit	No	Integer	Specifies the maximum number of flavors that can be displayed on one page. The default value is 1000 .
marker	No	String	Specifies the ID of the last flavor on one page as the paging marker.

Request

None

Response

[Table 4-96](#) describes the response parameters.

Table 4-96 Response parameters

Parameter	Mandatory	Type	Description
flavors	Yes	Array of objects	Specifies ECS flavors. For details, see Table 4-97 .

Table 4-97 flavors field description

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the ECS flavor ID.
name	Yes	String	Specifies the ECS flavor name.
vcpus	Yes	String	Specifies the number of vCPUs in the ECS flavor.
ram	Yes	Integer	Specifies the memory size (MB) in the ECS flavor.
disk	Yes	String	Specifies the system disk size in the ECS flavor. This parameter has not been used. Its default value is 0 .
swap	No	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	Yes	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	Yes	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 .

Parameter	Mandatory	Type	Description
rxtx_factor	Yes	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 .
rxtx_quota	Yes	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	Yes	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	Yes	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	Yes	Array of objects	Specifies the shortcut link of the ECS flavor. For details, see Table 4-98 .
extra_specs	Yes	Object	Specifies the extended field of the ECS specifications. For details, see Table 4-92 .

Table 4-98 links field description

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

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/resize_flavors?source_flavor_id=c3.xlarge.2
```

Example Response

```
{
  "flavors": [
    {
      "id": "c3.15xlarge.2",
      "name": "c3.15xlarge.2",
      "vcpus": "60",
      "ram": 131072,
      "disk": "0",
      "swap": "",
      "links": [
        {
          "rel": "self",
          "href": "https://ecs.region.xxx.com/v1.0/743b4c0428d94531b9f2add666642e6b/flavors/c3.15xlarge.2",
          "type": null
        },
        {
          "rel": "bookmark",
          "href": "https://ecs.region.xxx.com/743b4c0428d94531b9f2add666642e6b/flavors/c3.15xlarge.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,
      "extra_specs": {
        "ecs:virtualization_env_types": "CloudCompute",
        "ecs:generation": "c3",
        "ecs:performancetype": "computingv3",
        "resource_type": "IOOptimizedC3_2"
      }
    }
  ]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.4.3 Querying the Target Flavors to Which an ECS Flavor Can Be Changed (Discarded)

Function

An ECS flavor cannot be changed to certain flavors. This API is used to query the target flavors to which a specified ECS flavor can be changed.

This API has been discarded. Use the API described in [4.4.2 Querying the Target ECS Flavors to Which a Flavor Can Be Changed](#).

URI

GET /v2.1/{project_id}/resize_flavors?
instance_uuid={instance_uuid}&source_flavor_id={source_flavor_id}&source_flavor_name={source_flavor_name}&sort_key={sort_key}&sort_dir={sort_dir}&limit={limit}&marker={marker}

Table 4-99 describes the parameters in the URI.

Table 4-99 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

NOTE

One of the **instance_uuid**, **source_flavor_id**, and **source_flavor_name** parameters must be configured. If multiple parameters are configured, the system processes the **instance_uuid**, **source_flavor_id**, and **source_flavor_name** parameters in descending order by default.

Table 4-100 describes the query parameters.

Table 4-100 Query parameters

Parameter	Mandatory	Type	Description
instance_uuid	No	String	Specifies the ID, in UUID format, of the target ECS.
source_flavor_id	No	String	Specifies the source flavor ID.
source_flavor_name	No	String	Specifies the source flavor name.
sort_key	No	String	Indicates the field for sorting. Options: <ul style="list-style-type: none">● flavorid: indicates the flavor ID. The default value is flavorid.● name: indicates the flavor name.● memory_mb: indicates the memory size.● vcpus: indicates the number of vCPUs.● root_gb: indicates the system disk size.

Parameter	Mandatory	Type	Description
sort_dir	No	String	Specifies the ascending or descending sorting. Options: <ul style="list-style-type: none"> • asc: indicates the ascending order. • desc: indicates the descending order.
limit	No	Integer	Specifies the maximum number of flavors that can be displayed on one page. The default value is 1000 .
marker	No	String	Uses the ID of the last flavor on one page as the paging marker.

Request

None

Response

[Table 4-101](#) describes the response parameters.

Table 4-101 Response parameters

Parameter	Mandatory	Type	Description
flavors	Yes	Array of objects	Specifies ECS flavors. For details, see Table 4-102 .

Table 4-102 flavors field description

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

Parameter	Mandatory	Type	Description
swap	No	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	Yes	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	Yes	Boolean	This is an extended attribute, specifying whether a flavor is available. <ul style="list-style-type: none"> • true: indicates that a flavor is available. • false: indicates that a flavor is unavailable. NOTE This parameter is not used.
rxtx_factor	Yes	Float	This is an extended attribute. NOTE This parameter is not used.
rxtx_quota	Yes	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	Yes	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	Yes	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	Yes	Array of objects	Specifies the shortcut link of the ECS flavor. For details, see Table 4-103 .

Parameter	Mandatory	Type	Description
extra_specs	Yes	Array of objects	Specifies the extended field of the ECS specifications. For details, see Table 4-92 .

Table 4-103 links field description

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

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/resize_flavors?source_flavor_id=c3.xlarge.2
```

Example Response

```
{
  "flavors": [
    {
      "id": "c3.15xlarge.2",
      "name": "c3.15xlarge.2",
      "vcpus": "60",
      "ram": "131072",
      "disk": "0",
      "swap": "",
      "links": [
        {
          "rel": "self",
          "href": "https://compute-ext.region.xxx.com/v1.0/743b4c0428d94531b9f2add666642e6b/flavors/c3.15xlarge.2",
          "type": null
        },
        {
          "rel": "bookmark",
          "href": "https://compute-ext.region.xxx.com/743b4c0428d94531b9f2add666642e6b/flavors/c3.15xlarge.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,
      "extra_specs": {
        "ecs:virtualization_env_types": "CloudCompute",
        "ecs:generation": "c3",
        "ecs:performancetype": "computingv3",
        "resource_type": "IOptimizedC3_2"
      }
    }
  ]
}
```

```
}  
  ]  
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.5 NIC Management

4.5.1 Adding NICs to an ECS in a Batch

Function

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

URI

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

[Table 4-104](#) describes the parameters in the URI.

Table 4-104 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-105](#) describes the request parameters.

Table 4-105 Request parameters

Parameter	Mandatory	Type	Description
nics	Yes	Array of objects	Specifies the NICs to be added. For details, see Table 4-106 .

Table 4-106 nics field description

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	Specifies the information about the NICs to be added to an ECS.
security_groups	No	Array of objects	Specifies the security groups for NICs. For details, see Table 4-107 .
ip_address	No	String	Specifies the IP address. If this parameter is unavailable, the IP address is automatically assigned.

Table 4-107 security_groups field description

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

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

```
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

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.5.2 Deleting NICs from an ECS in a Batch

Function

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

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 4-108](#) describes the parameters in the URI.

Table 4-108 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-109](#) describes the request parameters.

Table 4-109 Request parameters

Parameter	Mandatory	Type	Description
nics	Yes	Array of objects	Specifies the NICs to be deleted. For details, see Table 4-110 .

Table 4-110 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 [9.5.1 Responses \(Task\)](#).

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.5.3 Binding a Virtual IP Address to an ECS NIC

Function

A virtual IP address provides the second IP address for one or multiple ECS NICs, improving high availability between the ECSs.

This API is used to configure a virtual IP address for an ECS NIC.

- If the specified IP address is a virtual IP address that has not been assigned, the system automatically assigns the virtual IP address and binds it to a specified NIC.
- If the specified IP address is a virtual IP address that has been assigned, the system binds the virtual IP address to a specified NIC. If the **device_owner** of this IP address is left blank, only intra-VPC layer 2 and layer 3 communication is supported. If the **device_owner** of this IP address is **neutron:VIP_PORT**, intra-VPC layer 2 and layer 3 communication, inter-VPC peer access, as well as Internet access through EIP, VPN, and Cloud Connect are supported.

URI

PUT /v1/{project_id}/cloudservers/nics/{nic_id}

[Table 4-111](#) describes the parameters in the URI.

Table 4-111 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
nic_id	Yes	Specifies the ECS NIC ID.

Request

[Table 4-112](#) describes the request parameters.

Table 4-112 Request parameters

Parameter	Mandatory	Type	Description
nic	Yes	Object	Specifies the NIC parameters required for binding a virtual IP address. For details, see Table 4-113 .

Table 4-113 nic field description

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	Specifies the information about the NICs to be added to an ECS. Set the parameter value to the ID (in UUID format) of the network created in the VPC to which the target ECS belongs.
ip_address	Yes	String	Specifies the virtual IP address to be bound to a NIC.
reverse_binding	No	Boolean	Indicates the allowed_address_pairs attribute of a virtual IP address, specifying whether the NIC IP/MAC address pair is added.

Response

[Table 4-114](#) describes the response parameters.

Table 4-114 Response parameters

Parameter	Type	Description
port_id	String	Specifies the ECS NIC ID.

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/nics/{nic_id}
{
  "nic": {
    "subnet_id": "d32019d3-bc6e-4319-9c1d-6722fc136a23",
    "ip_address": "192.168.0.7",
    "reverse_binding": true
  }
}
```

Example Response

```
{
  "port_id": "d32019d3-bc6e-4319-9c1d-6722fc136a23"
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.5.4 Unbinding a Virtual IP Address from an ECS NIC

Function

A virtual IP address provides the second IP address for one or multiple ECS NICs, improving high availability between the ECSs.

This API is used to unbind a virtual IP address from an ECS NIC. After the NIC is unbound, it is not deleted. For instructions about how to delete an ECS NIC, see [4.5.2 Deleting NICs from an ECS in a Batch](#).

URI

```
PUT /v1/{project_id}/cloudservers/nics/{nic_id}
```

[Table 4-115](#) describes the parameters in the URI.

Table 4-115 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
nic_id	Yes	Specifies the ECS NIC ID.

Request

[Table 4-116](#) describes the request parameters.

Table 4-116 Request parameters

Parameter	Mandatory	Type	Description
nic	No	Object	When this parameter is specified, it is the NIC required for unbinding a virtual IP address. For details, see Table 4-117 . When this parameter is not specified, Source/Destination Check is enabled. In this case, the system checks whether the source IP address in the packets sent by the ECS is correct. If the source IP address is incorrect, the packets cannot be sent.

Table 4-117 nic field description

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	Specifies the information about the NICs to be added to an ECS. This parameter must be left blank when you unbind the virtual IP address from an ECS NIC.
ip_address	Yes	String	Specifies the virtual IP address to be unbound from a NIC. This parameter must be left blank when you unbind the virtual IP address from an ECS NIC.

Parameter	Mandatory	Type	Description
reverse_binding	No	Boolean	Indicates the allowed_address_pairs attribute of a virtual IP address, specifying whether the NIC IP/MAC address pair is added.

Response

[Table 4-118](#) describes the response parameters.

Table 4-118 Response parameters

Parameter	Type	Description
port_id	String	Specifies the ECS NIC ID.

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/nics/{nic_id}
{
  "nic": {
    "subnet_id": "",
    "ip_address": "",
    "reverse_binding": false
  }
}
```

Example Response

```
{
  "port_id": "d32019d3-bc6e-4319-9c1d-6722fc136a23"
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.5.5 Query NICs of an ECS

Function

This API is used to query NICs of an ECS.

URI

GET /v1/{project_id}/cloudservers/{server_id}/os-interface

[Table 4-119](#) describes the parameters in the URI.

Table 4-119 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-120](#) describes the response parameters.

Table 4-120 Response parameters

Parameter	Type	Description
interfaceAttachments	Array of objects	Specifies ECS NICs. For details, see Table 4-121 .

Table 4-121 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 4-122 .
net_id	String	Specifies the network ID to which the NIC port belongs.
port_id	String	Specifies the NIC port ID.
mac_addr	String	Specifies the MAC address of the NIC.

Table 4-122 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

```
GET https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/os-interface
```

Example Response

```
{
  "interfaceAttachments": [
    {
      "port_state": "ACTIVE",
      "fixed_ips": [
        {
          "subnet_id": "ba31e1f5-fa76-4530-862c-5176fad033cf",
          "ip_address": "192.168.0.33"
        }
      ],
      "net_id": "610a4af2-1d90-4d2b-8057-dc238b26feb",
      "port_id": "04819c0a-6a07-44b6-945e-fb932071888e",
      "mac_addr": "fa:16:3e:45:65:c4"
    }
  ]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

4.6 Disk Management

4.6.1 Querying a Single Disk Attached to an ECS

Function

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

URI

```
GET /v1/{project_id}/cloudservers/{server_id}/block_device/{volume_id}
```

[Table 4-123](#) describes the parameters in the URI.

Table 4-123 Parameter description

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID in UUID format.
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
volume_id	Yes	Specifies the EVS disk ID in UUID format.

Request

None

Response

[Table 4-124](#) describes the response parameters.

Table 4-124 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disk attached to an ECS. For details, see Table 4-125 .

Table 4-125 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.
bootIndex	Integer	Specifies the EVS disk boot sequence. <ul style="list-style-type: none">● 0 indicates the system disk.● Non-0 indicates a data disk.
bus	String	Specifies the disk bus type. Options: virtio and scsi

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/{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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.6.2 Querying Disk Attachments of an ECS

Function

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

URI

GET /v1/{project_id}/cloudservers/{server_id}/os-volume_attachments

[Table 4-126](#) describes the parameters in the URI.

Table 4-126 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID in UUID format.

Request

None

Response

[Table 4-127](#) describes the response parameters.

Table 4-127 Response parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies disks attached to an ECS. For details, see Table 4-128 .

Table 4-128 volumeAttachments field description

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

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/{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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.6.3 Querying Information About Disks Attached to an ECS

Function

This API is used to query information about disks attached to an ECS.

URI

GET /v1/{project_id}/cloudservers/{server_id}/block_device

[Table 4-129](#) describes the parameters in the URI.

Table 4-129 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID in UUID format.

Request

None

Response

[Table 4-130](#) describes the response parameters.

Table 4-130 Response parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies the disks attached to an ECS. For details, see Table 4-131 .
attachableQuantity	Object	Specifies the number of disks that can be attached to an ECS. For details, see Table 4-132 .

Table 4-131 volumeAttachments parameters

Parameter	Type	Description
serverId	String	Specifies the ECS ID in UUID format.
volumeId	String	Specifies the EVS disk ID in UUID format.

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

Table 4-132 attachableQuantity parameters

Parameter	Type	Description
free_scsi	Integer	Specifies the number of SCSI disks that can be attached to an ECS.
free_blk	Integer	Specifies the number of virtio_blk disks that can be attached to an ECS.
free_disk	Integer	Specifies the total number of disks that can be attached to an ECS.

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/block_device
```

Example Response

```
{
  "attachableQuantity": {
    "free_scsi": 23,
    "free_blk": 15,
    "free_disk": 23
  },
  "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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.6.4 Attaching a Disk to an ECS

Function

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

URI

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

[Table 4-133](#) describes the parameters in the URI.

Table 4-133 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

Table 4-134 Request parameters

Parameter	Mandatory	Type	Description
volumeAttachment	Yes	Object	Specifies the disk attached to an ECS.

Table 4-135 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 <i>x</i> 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.

Response

See [9.5.1 Responses \(Task\)](#).

Example Request

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/attachvolume
{
  "volumeAttachment": {
    "volumeld": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
    "device": "/dev/sda"
  }
}
```

Example Response

```
{
  "job_id": "70a599e0-31e7-49b7-b260-868f441e862b"
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.6.5 Detaching an EVS Disk from an ECS

Function

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

URI

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

[Table 4-136](#) describes the parameters in the URI.

Table 4-136 Parameter description

Parameter	Mandator y	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 [9.5.1 Responses \(Task\)](#).

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.6.6 Querying Disk Attachment of an ECS (Discarded)

Function

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

This API has been discarded. Use the API described in [4.6.2 Querying Disk Attachments of an ECS](#).

URI

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

[Table 4-137](#) describes the parameters in the URI.

Table 4-137 Parameter description

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

Request

None

Response

[Table 4-138](#) describes the response parameters.

Table 4-138 Response parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies the disks attached to an ECS. For details, see Table 4-139 .
attachableQuantity	Object	Specifies the number of disks that can be attached to an ECS. For details, see Table 4-140 .

Table 4-139 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.
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.• Non-0 indicates a data disk.
bus	String	Indicates the disk bus type. Options: virtio and scsi

Table 4-140 attachableQuantity parameters

Parameter	Type	Description
free_scsi	Integer	Specifies the number of SCSI disks that can be attached to an ECS.
free_blk	Integer	Specifies the number of virtio_blk disks that can be attached to an ECS.
free_disk	Integer	Specifies the total number of disks that can be attached to an ECS.

Example Request

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

Example Response

```
{
  "attachableQuantity": {
    "free_scsi": 23,
    "free_blk": 15,
    "free_disk": 23
  },
  "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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.6.7 Querying a Single Disk Attached to an ECS (Discarded)

Function

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

This API has been discarded. Use the API described in [4.6.1 Querying a Single Disk Attached to an ECS](#).

URI

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

[Table 4-141](#) describes the parameters in the URI.

Table 4-141 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 4-142](#) describes the response parameters.

Table 4-142 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disk attached to an ECS. For details, see Table 4-143 .

Table 4-143 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.
bootIndex	Boolean	Specifies the EVS disk boot sequence. <ul style="list-style-type: none">• 0 indicates the system disk.• Non-0 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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.7 Metadata Management

4.7.1 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 /v1/{project_id}/cloudservers/{server_id}/metadata

Table 4-144 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

Table 4-145 Request parameters

Parameter	Mandatory	Type	Description
metadata	Yes	Object	<p>Specifies the user-defined metadata key-value pair.</p> <p>The data structure can be empty. If the value is empty, data is not updated.</p> <p>For a metadata tag:</p> <p>It contains a maximum of 255 Unicode characters and cannot be left blank. A tag can contain uppercase letters (A-Z), lowercase letters (a-z), digits (0-9), hyphens (-), underscores (_), colons (:), and periods (.).</p> <p>For a metadata value:</p> <p>It contains a maximum of 255 Unicode characters.</p>

Response

Table 4-146 Parameter description

Parameter	Type	Description
metadata	Object	Specifies the user-defined metadata key-value pair.

Example Request

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/metadata
{
  "metadata": {
    "key": "value"
  }
}
```

Example Response

```
{
  "metadata": {
    "key": "value"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

4.7.2 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 /v1/{project_id}/cloudservers/{server_id}/metadata/{key}

[Table 4-147](#) describes the parameters in the URI.

Table 4-147 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
key	Yes	Specifies the ECS metadata key value to be deleted.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/metadata/{key}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

4.8 Tenant Quota Management

4.8.1 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 4-148](#) describes the parameters in the URI.

Table 4-148 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 4-149](#) describes the response parameters.

Table 4-149 Response parameters

Parameter	Type	Description
absolute	Object	Specifies tenant quotas. For details, see Table 4-150 .

Table 4-150 absolute field description

Parameter	Type	Description
maxTotalInstances	Integer	Specifies the maximum number of ECSs you can use.
maxTotalCores	Integer	Specifies the maximum number of CPU cores you can use.
maxTotalRAMSize	Integer	Specifies the maximum memory space (MB) you can use.

Parameter	Type	Description
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.
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.
maxImageMetadata	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

```
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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.9 Task Status Management

4.9.1 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 [9.5.1 Responses \(Task\)](#).

URI

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

[Table 4-151](#) describes the parameters in the URI.

Table 4-151 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
job_id	Yes	Specifies the ID of an asynchronous request task.

Request

None

Response

[Table 4-152](#) describes the response parameters.

Table 4-152 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.• PENDING_PAYMENT: indicates that a yearly/monthly order is to be paid. NOTE The PENDING_PAYMENT status is displayed after the request for creating a yearly/monthly ECS or modifying the specifications of yearly/monthly ECS has been submitted and before the order is paid. If the order is canceled, the status will not be automatically updated. The task will be automatically deleted 14 days later.
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 4-153 .
job_id	String	Specifies the ID of an asynchronous request task.

Parameter	Type	Description
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 9.1 Returned Values for General Requests .

Table 4-153 entities field description

Parameter	Type	Description
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 4-154 .

Table 4-154 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.

Parameter	Type	Description
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 4-155 .
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 4-155 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

```
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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.10 Tag Management

4.10.1 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 section [5.16 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.

4.10.2 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}/cloudservers/{server_id}/tags/action

[Table 4-156](#) describes the parameters in the URI.

Table 4-156 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-157](#) describes the request parameters.

Table 4-157 Request parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of objects	Specifies tags. For details, see Table 4-158 .
action	Yes	String	Specifies the operation. (Only lowercase letters are supported.) For example, create indicates the creation operation.

Table 4-158 tags 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.• Contains a maximum of 36 characters.• Must be unique and cannot be left blank.

Parameter	Mandatory	Type	Description
value	Yes	String	Specifies the tag value. <ul style="list-style-type: none">Contains a maximum of 43 characters.

Response

None

Example Request

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/tags/action
{
  "action": "create",
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value3"
    }
  ]
}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.10.3 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.

NOTE

- 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}/cloudservers/{server_id}/tags/action

[Table 4-159](#) describes the parameters in the URI.

Table 4-159 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-160](#) describes the request parameters.

Table 4-160 Request parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of objects	Specifies tags. For details, see Table 4-161 .
action	Yes	String	Specifies the operation. (Only lowercase letters are supported.) For example, delete indicates the deletion operation.

Table 4-161 tags 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. It contains a maximum of 255 Unicode characters and can be left blank.

Response

None

Example Request

```
POST https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/tags/action
{
  "action": "delete",
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value3"
    }
  ]
}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.10.4 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 under one account.

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

URI

GET /v1/{project_id}/cloudservers/tags

[Table 4-162](#) describes the parameters in the URI.

Table 4-162 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 4-163](#) describes the response parameters.

Table 4-163 Response parameters

Parameter	Type	Description
tags	Array of objects	Specifies tags.

Table 4-164 tag field description

Parameter	Type	Description
key	String	Specifies the tag key. <ul style="list-style-type: none">Contains a maximum of 36 Unicode characters.
values	Array of strings	Specifies the tag value. <ul style="list-style-type: none">Contains a maximum of 43 Unicode characters.Can be left blank.

Example Request

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

Example Response

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

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.10.5 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}/cloudservers/{server_id}/tags

[Table 4-165](#) describes the parameters in the URI.

Table 4-165 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-166](#) describes the response parameters.

Table 4-166 Response parameters

Parameter	Type	Description
tags	Array of objects	Specifies tags. For details, see Table 4-167 .

Table 4-167 tags 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}/cloudservers/{server_id}/tags
```

Example Response

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

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.10.6 Querying Tags of an ECS (Discarded)

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.

NOTE

This API has been discarded. Use the API described in [4.10.5 Querying Tags of an ECS](#).

URI

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

[Table 4-168](#) describes the parameters in the URI.

Table 4-168 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-169](#) describes the response parameters.

Table 4-169 Response parameters

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

Table 4-170 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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.11 Password Management

4.11.1 Querying Whether One-Click Password Reset Is Supported

Function

This API is used to query whether one-click password reset is supported.

URI

GET /v1/{project_id}/cloudservers/{server_id}/os-resetpwd-flag

[Table 4-171](#) describes the parameters in the URI.

Table 4-171 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-172](#) describes the response parameters.

Table 4-172 Response parameters

Parameter	Type	Description
resetpwd_flag	String	Specifies whether one-click password reset is supported. <ul style="list-style-type: none">• True: One-click password reset is supported.• False: One-click password reset is not supported.

Example Request

```
GET https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/os-resetpwd-flag
```

Example Response

```
{  
  "resetpwd_flag": "False"  
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.11.2 Resetting the Password for Logging In to an ECS with a Few Clicks

Function

This API is used to reset the password of the ECS management account, **root** or **Administrator**.

Constraints

- By default, there is no password complexity check that meets security requirements. No error message is displayed after an insecure password is entered.
- Before using this API, you must install password reset plug-ins. For instructions about how to download and install the password reset plug-ins, see "Installing One-Click Password Reset Plug-ins" in *Elastic Cloud Server User Guide*.
- You cannot determine whether an ECS supports password reset.
- If the password reset function fails to take effect, this API does not report an error.
- A new password takes effect after the ECS is started or restarted.

URI

PUT /v1/{project_id}/cloudservers/{server_id}/os-reset-password

[Table 4-173](#) lists the URI parameters.

Table 4-173 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 4-174](#) describes the request parameters.

Table 4-174 Request parameters

Parameter	Mandatory	Type	Description
reset-password	Yes	Object	Provides reset-password details.

Table 4-175 reset-password field description

Parameter	Mandatory	Type	Description
is_check_password	No	Boolean	Specifies whether to check password complexity.
new_password	Yes	String	<p>Specifies the new password for logging in to an ECS.</p> <p>By default, this API does not check password security. To check password security, set is_check_password to true.</p> <p>A new password must comply with the following rules:</p> <ul style="list-style-type: none"> • Consists of 8 to 26 characters. • Supports the following characters: !@%_-=+[]:./? • Cannot contain Chinese characters or the following characters: ¥— — ~`#&^,{}*();'"<> \ \$ • Contains at least three of the following: uppercase letters, lowercase letters, digits, and allowed special characters. • Cannot contain username Administrator/root or the username spelled backwards. • Cannot contain three consecutive characters in username Administrator

Response

None

Example Request

```
PUT https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/os-reset-password
{
  "reset-password": {
    "new_password": "Fusion@123"
  }
}
```



```
}  
}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.11.3 Retrieving the Password for Logging In to a Windows ECS

Function

This API is used to obtain the random password generated for user **Administrator** or the user configured in Cloudbase-Init when you use a Cloudbase-Init-enabled image to create a Windows ECS.

URI

GET /v1/{project_id}/cloudservers/{server_id}/os-server-password

[Table 4-176](#) lists the URI parameters.

Table 4-176 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 4-177](#) describes the response parameters.

Table 4-177 Response parameters

Parameter	Type	Description
password	String	Specifies the password in ciphertext.

Example Request

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

Example Response

```
{  
  "password": "UHC9+YW1xDC1Yu8Mg9n+tnOp7euEO/cW//9KgdJKWhr5w=="  
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

4.11.4 Deleting the Password for Logging In to a Windows ECS

Function

This API is used to delete the recorded random password generated during initial Windows ECS installation. 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.

URI

```
DELETE /v1/{project_id}/cloudservers/{server_id}/os-server-password
```

[Table 4-178](#) lists the URI parameters.

Table 4-178 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v1/{project_id}/cloudservers/{server_id}/os-server-password
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

4.11.5 Resetting the Password for Logging In to an ECS with a Few Clicks (Discarded)

Function

This API is used to reset the password of the ECS management account, **root** or **Administrator**.

This API has been discarded. Use the API described in [4.11.2 Resetting the Password for Logging In to an ECS with a Few Clicks](#).

Constraints

- There is no password complexity check that meets security requirements. No error message is displayed after an insecure password is entered.
- Before using this API, you must install password reset plug-ins. For instructions about how to download and install the password reset plug-ins, see "Installing One-Click Password Reset Plug-ins" in *Elastic Cloud Server User Guide*.
- This API cannot detect whether the target ECS supports password reset.
- If resetting the password for logging in to an ECS failed, this API will not report an error.
- A new password takes effect after the ECS is started or restarted.

URI

PUT /v2.1/{project_id}/servers/{server_id}/os-reset-password

[Table 4-179](#) describes the parameters in the URI.

Table 4-179 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

Table 4-180 describes the request parameters.

Table 4-180 Request parameters

Parameter	Type	Mandatory	Description
reset-password	Object	Yes	Provides reset-password details.

Table 4-181 reset-password field description

Parameter	Type	Mandatory	Description
new_password	String	Yes	<p>Specifies the new password for logging in to the ECS.</p> <p>This API does not check password security. Ensure that the password complexity complies with the password rules.</p> <p>The password rules are as follows:</p> <ul style="list-style-type: none">• Consists of 8 to 26 characters.• Must contain at least three of the following character types:<ul style="list-style-type: none">- Uppercase letters- Lowercase letters- Digits- Special characters, including !@\$%^&*_+=[]:;./?• Cannot contain the username or the username in reverse.• Cannot contain more than two characters in the same sequence as they appear in the username. (This requirement applies only to Windows ECSs.)

Response

None

Example Request

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-reset-password
```

```
{
  "reset-password": {
    "new_password": "Fusion@123"
  }
}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.12 ECS Group Management

4.12.1 Creating an ECS Group

Function

This API is used to create an ECS group.

Constraints

Only anti-affinity policies are supported.

URI

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

[Table 4-182](#) describes the parameters in the URI.

Table 4-182 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 4-183](#) describes the request parameters.

Table 4-183 Request parameters

Parameter	Mandatory	Type	Description
server_group	Yes	Object	Specifies the ECS group information.

Table 4-184 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	<p>Specifies the policies associated with the ECS group. Options:</p> <ul style="list-style-type: none">● anti-affinity: ECSs in this group must be deployed on different hosts.● affinity: ECSs in this group must be deployed on the same host.● soft-anti-affinity: ECSs in this group are deployed on different hosts if possible. If the ECSs cannot be deployed on different hosts, deploy them based on the actual condition for successful ECS creation.● soft-affinity: ECSs in this group are deployed on the same host if possible. If the ECSs cannot be deployed on the same host, deploy them based on the actual condition for successful ECS creation. <p>NOTE Only anti-affinity policies are supported. You are not advised to use other policies. If other policies are used, creating the ECS group will fail.</p>

Response

[Table 4-185](#) describes the response parameters.

Table 4-185 Response parameters

Parameter	Type	Description
server_group	Object	Specifies the ECS group information.

Table 4-186 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.• affinity: ECSs in this group must be deployed on the same host.• soft-anti-affinity: ECSs in this group are deployed on different hosts if possible. If the ECSs cannot be deployed on different hosts, deploy them based on the actual condition for successful ECS creation.• soft-affinity: ECSs in this group are deployed on the same host if possible. If the ECSs cannot be deployed on the same host, deploy them based on the actual condition for successful ECS creation.
members	Array of strings	Specifies the IDs of the ECSs in an ECS group.
metadata	Object	Specifies the ECS group metadata.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.12.2 Deleting an ECS Group

Function

This API is used to delete an ECS group.

URI

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

[Table 4-187](#) describes the parameters in the URI.

Table 4-187 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group UUID.

Request Parameters

None

Response Parameters

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.12.3 Querying ECS Groups

Function

This API is used to query ECS groups.

URI

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

[Table 4-188](#) describes the parameters in the URI.

Table 4-188 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 4-189](#) describes the response parameters.

Table 4-189 Response parameters

Parameter	Type	Description
server_groups	Array of objects	Specifies ECS groups.

Table 4-190 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	<p>Specifies the policies associated with the ECS group. Options:</p> <ul style="list-style-type: none">● anti-affinity: ECSs in this group must be deployed on different hosts.● affinity: ECSs in this group must be deployed on the same host.● soft-anti-affinity: ECSs in this group are deployed on different hosts if possible. If the ECSs cannot be deployed on different hosts, deploy them based on the actual condition for successful ECS creation.● soft-affinity: ECSs in this group are deployed on the same host if possible. If the ECSs cannot be deployed on the same host, deploy them based on the actual condition for successful ECS creation. <p>NOTE Only anti-affinity policies are supported. You are not advised to use other policies. If other policies are used, creating the ECS group will fail.</p>

Example Request

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

Example Response

```
{
  "server_groups": [
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "name": "test",
      "policies": ["anti-affinity"],
      "members": [],
      "metadata": {}
    }
  ]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

4.12.4 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 4-191](#) describes the parameters in the URI.

Table 4-191 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_group_id	Yes	Specifies an ECS group UUID.

Request

None

Response

[Table 4-192](#) describes the response parameters.

Table 4-192 Response parameters

Parameter	Type	Description
server_group	Object	Specifies an ECS group.

Table 4-193 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

```
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 [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

5 OpenStack Nova APIs

5.1 API Version Query

5.1.1 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 5-1 Response parameters

Parameter	Type	Description
versions	Object	Specifies the API versions. For details, see Table 5-2 .

Table 5-2 versions field description

Parameter	Type	Description
id	string	Specifies the version ID.
links	Object	Specifies shortcut links for versions. For details, see Table 5-3 .
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	<p>Specifies the API version status. Possible values are as follows:</p> <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	<p>The value of this parameter varies by API version.</p> <p>If the API version is 2.0, the value is 2011-01-21T11:33:21Z.</p> <p>If the API version is 2.1, the value is 2013-07-23T11:33:21Z.</p>

Table 5-3 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

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

Example Response

```
{
  "versions": [{
    "links": [{
      "rel": "self",
      "href": "https://ecs.service.domain.com:443/v2/"
    }],
    "id": "v2.0",
    "updated": "2001-09-21T12:33:21Z",
    "status": "SUPPORTED"
  }]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.1.2 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 5-4](#) describes the parameters in the URI.

Table 5-4 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 5-5 Response parameters

Parameter	Type	Description
versions	Object	Specifies the versions. For details, see Table 5-6 .

Table 5-6 versions field description

Parameter	Type	Description
id	string	Specifies the version ID.

Parameter	Type	Description
links	Object	Specifies the links to resources. For more information, see the OpenStack Documentation . For details, see Table 5-7 .
media-types	Object	Specifies the media types. For details, see Table 5-8 .
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.
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 .

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.

Table 5-7 links field description

Parameter	Type	Description
href	string	Specifies the links of the corresponding resources.
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 5-8 media-types field description

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

Example Request

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

Example Response

```
{
  "version": {
    "min_version": "",
    "media-types": [{
      "type": "application/vnd.openstack.compute+json;version=2",
      "base": "application/json"
    }],
    "links": [{
      "rel": "self",
      "href": "https://ecs.service.domain.com:443/v2/"
    }],
    {
      "rel": "describedby",
      "href": "http://docs.openstack.org/",
      "type": "text/html"
    }
  ],
  "id": "v2.0",
  "updated": "1999-02-20T11:33:21Z",
  "version": "",
  "status": "SUPPORTED"
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.2 Lifecycle Management

5.2.1 Creating an ECS

Function

This API is used to create a pay-per-use 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 [4.1.2 Creating an ECS \(Pay-per-Use\)](#).

URI

POST /v2.1/{project_id}/servers

[Table 5-9](#) describes the parameters in the URI.

Table 5-9 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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, use the ECS APIs described in [4.1.2 Creating an ECS \(Pay-per-Use\)](#).
2. 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.
3. A file injection failure will result in the ECS creation failure.
4. 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.
5. Native APIs `/v2/{project_id}/servers` and `/v2.1/{project_id}/servers` provided by the public 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 public cloud platform: creates an ECS using the shared storage as the system disk.

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

- a. You can query information about the 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.
6. When you create an ECS with a specified disk, ensure that the disk and the ECS are in the same AZ.

- 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.
- ECSs cannot be created in networks with **provider:network_type** set to **geneve**.

 **NOTE**

If **provider:network_type** is set to **geneve**, the internal high-speed network is used for BMSs.

- During the ECS creation, multiple ports can belong to the same network only when the global configuration item **allow_duplicate_networks** is enabled.
- 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.

Request

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

Table 5-10 Request parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the ECS information. For details, see Table 5-11 .
os:scheduler_hints	No	Object	Specifies the ECS scheduling information. For details, see Table 5-16 . This parameter is not available in BMS scenarios.

Table 5-11 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-b8b7-5a7f0f0af8f2• Example image URL: http://glance.openstack.example.com/images/3b8d6fef-af77-42ab-b8b7-5a7f0f0af8f2• 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: c1.2xlarge
name	Yes	String	Specifies the ECS name. The value contains 1 to 255 characters. NOTE ECS hostnames comply with RFC952 and RFC1123 naming rules. It is recommended that you configure hostnames using digits, letters (case sensitive), and hyphens (-). Underscores (_) are converted into hyphens (-) by default.
metadata	No	Object	Specifies the ECS metadata. For details, see Table 5-12 . <ul style="list-style-type: none">• The key contains 1 to 255 characters.• The value contains 0 to 255 characters.
adminPass	No	String	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 .

Parameter	Mandatory	Type	Description
block_device_mapping_v2	No	Array of objects	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 5-13 . This parameter is not available in BMS scenarios.
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 5-14 .
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 5-15 .
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 during the ECS 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 <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 lyEgL2Jpbi9iYXNoDQplY2hvlHVzZXJfdGVzd-CAmZ3Q7Jmd0OyAvaG9tZS91c2VyLnR4dA== Windows cmVtIGNtZA0KZWNoYyAxMTEgJmd0OyBjOlxhYWEudHh0
availability_zone	No	String	<p>Specifies the AZ of a specified ECS. This is an extended attribute.</p> <p>This parameter is mandatory when you create an ECS.</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 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: The max_count value must be greater than or equal to the min_count value.</p> <p>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> <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>

Parameter	Mandatory	Type	Description
description	No	String	<p>Specifies the description of an ECS, which is a null string by default. This is an extended attribute.</p> <ul style="list-style-type: none"> • Can contain a maximum of 85 characters. • Cannot contain special characters, such as < and >. <p>NOTE</p> <ul style="list-style-type: none"> • This field is supported in microversions 2.19 and later.

Table 5-12 metadata field description

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

Table 5-13 block_device_mapping_v2 parameters

Parameter	Type	Mandatory	Description
source_type	String	Yes	<p>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.</p> <p>NOTE</p> <p>If source_type is snapshot and boot_index is 0, the EVS disk of this snapshot must be the system disk.</p>

Parameter	Type	Mandatory	Description
destination_type	String	No	Specifies the target type of the disk device. Its value can only be volume . 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.
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 . true : When an ECS is deleted, its disks are deleted. false : When an ECS is deleted, its disks are not deleted.
boot_index	String	Yes	Specifies whether it is a boot disk. 0 specifies a boot disk, and -1 specifies a non-boot disk. NOTE If source_type of the volume device is volume , there must be one boot_index whose value is 0 .
uuid	String	No	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.

Parameter	Type	Mandatory	Description
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 . For details, see Disk Types .

Table 5-14 security_groups parameters

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

Table 5-15 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 5-16 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. You are not advised to use other policies.
different_host	No	Array of strings	The function has not been supported, and this field is reserved.
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	Specifies whether the ECS is created on a Dedicated Host (DeH) or in a shared pool (default). The value can be shared or dedicated . <ul style="list-style-type: none">• shared: indicates the shared pool.• dedicated: indicates the DeH. The parameter value also takes effect for ECS query operations.

Parameter	Mandatory	Type	Description
dedicated_host_id	No	String	Specifies the DeH ID. This parameter takes effect only when the value of tenancy is dedicated . If you do not specify this parameter, the system will automatically assign a DeH to you to deploy ECSs. The parameter value also takes effect for ECS query operations.

Response

[Table 5-17](#) describes the response parameters.

Table 5-17 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 5-18 .

Table 5-18 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 5-19 .
security_groups	Array of objects	Specifies the security groups to which the ECS belongs. For details, see Table 5-20 .
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.

Parameter	Type	Description
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 5-19 links field description

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

Table 5-20 security_groups field description

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

Example Request (Creating an ECS)

Example URL request

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

Example 1: Use an image to create an ECS through the API block_device_mapping_v2.

```
{
  "server": {
    "flavorRef": "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":
"ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBrbm93IHdoeSBpdCBtb3ZlcyBpbjBqdXN0IHN1Y2ggYSBkaXJlY3Rp
b24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLkl0IGZlZWxzIGFuIGltcHVsc2lvbi4uLnRoXMGaXMgdGhliHBSYWNlIH
RvIGdvlG5vdy4gQnV0IHRoZSBza3kga25vd3MgdGhliHJlYXNvbnMgYW5kIHRoZSBwYXR0ZXJucyBiZWWhpbmQg
YWxsIGNsb3VkcycgYW5kIHlvdSB3aWxslGtub3csiHRvbywgd2hlbiB5b3UgbGlmdCB5b3Vyc2VsZiBoaWdoIGVu
b3VnaCB0byBzZWUgYmV5b25kIGhvcml6b25zLiINCg0KLVJpY2hhcmQgQmFjaA==",
"availability_zone": "az1-dc1"
}
}

```

Example 2: Use a snapshot to create an ECS through the API `block_device_mapping_v2`.

NOTE

When **source_type** is **snapshot**, **boot_index** is **0**, and the EVS disk corresponding to the snapshot must be a system disk.

```

{
  "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 returned
        "destination_type": "volume"
      }
    ],
    "flavorRef": "s3.xlarge.2",
    "max_count": 1,
    "min_count": 1,
    "networks": [
      {
        "uuid": "79a68cef-0936-4e21-b1f4-b800ecb70246"
      }
    ]
  }
}

```

Example 3: Use a disk to create an ECS through the API `block_device_mapping_v2`.

```

{
  "server": {
    "flavorRef": "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":
    "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBrbm93IHdoeSBpdCBtb3ZlcyBpbiBqdXN0IHN1Y2ggYSBkaXJlY3Rpb24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLkl0IGZlZWxzIGFuIGltcHVsc2lvbi4uLnRoaXMgaXMgdGhllHBsYWNIH RvIGdvIG5vdy4gQnV0IHRob2ZSBza3kga25vd3MgdGhllHJlYXNvbnMgYW5kIHRob2ZSBwYXR0ZXJucyBiZW hpbmQg YWxsIGNs b3VkcywgYW5kIHlvdSB3aWxsIGtub3csIH Rvbywg d2h1biB5b3UgbGlmdCB5b3Vyc2VsZiBoaWdoiGVu b3VnaCB0byBzZWUgYmV5b25kIGhvcml6b25zLiINCg0KLVJpY2hhcmQgQmFjaA==",
      "availability_zone": "az1-dc1"
    }
  }
}

```

Example 4: Create an ECS through the API imageRef.

```

{
  "server": {
    "flavorRef": "2",
    "name": "wjvm48",
    "metadata": {
      "name": "name_xx1",
      "id": "id_xxxx1"
    },
    "adminPass": "name_xx1",
    "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":
    "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBrbm93IHdoeSBpdCBtb3ZlcyBpbiBqdXN0IHN1Y2ggYSBkaXJlY3Rpb24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLkl0IGZlZWxzIGFuIGltcHVsc2lvbi4uLnRoaXMgaXMgdGhllHBsYWNIH RvIGdvIG5vdy4gQnV0IHRob2ZSBza3kga25vd3MgdGhllHJlYXNvbnMgYW5kIHRob2ZSBwYXR0ZXJucyBiZW hpbmQg YWxsIGNs b3VkcywgYW5kIHlvdSB3aWxsIGtub3csIH Rvbywg d2h1biB5b3UgbGlmdCB5b3Vyc2VsZiBoaWdoiGVu b3VnaCB0byBzZWUgYmV5b25kIGhvcml6b25zLiINCg0KLVJpY2hhcmQgQmFjaA==",
      "availability_zone": "az1-dc1"
    }
  }
}

```

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://192.168.82.230:8774/v2/dc4059e8e7994f2498b514ca04cdf44/servers/567c1557-0eca-422c-bfce-149d6b8f1bb8",
        "rel": "self"
      },
      {
        "href": "http://192.168.82.230:8774/dc4059e8e7994f2498b514ca04cdf44/servers/567c1557-0eca-422c-bfce-149d6b8f1bb8",
        "rel": "bookmark"
      }
    ]
  }
}

```

```
    ],  
    "adminPass": "name_xx1"  
  }  
}
```

Example Request (Creating ECSs in a Batch)

```
{  
  "server": {  
    "availability_zone": "az1.dc1",  
    "name": "test",  
    "imageRef": "10ff4f01-35b6-4209-8397-359cb4475fa0",  
    "flavorRef": "s1.medium",  
    "return_reservation_id": "true",  
    "networks": [  
      {  
        "uuid": "51bead38-d1a3-4d08-be20-0970c24b7cab"  
      }  
    ],  
    "min_count": "2",  
    "max_count": "3"  
  }  
}
```

Example Response (Creating ECSs in a Batch)

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

Returned Values

See [9.1 Returned Values for General Requests](#).

5.2.2 Modifying ECS Information

Function

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

URI

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

[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 9.4 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
server	Yes	Object	Specifies the ECS data structure. For details, see Table 5-23 .

Table 5-23 server field description

Parameter	Mandatory	Type	Description
name	No	String	Specifies the name of the modified ECS. The length is greater than 0 and less than 256
description	No	String	Describes the ECS. The value contains a maximum of 255 bytes. This field is supported in microversions 2.19 and later.

Response

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

Table 5-24 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 5-25 .

Table 5-25 server field description

Parameter	Type	Description
tenant_id	String	Specifies the tenant or project ID.
image	String	Specifies the image ID.
accessIPv4	String	Reserved

Parameter	Type	Description
addresses	Object	Specifies the network addresses of an ECS. For details, see Table 5-26 .
metadata	Object	Specifies the ECS metadata.
accessIPv6	String	Reserved
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 ECS flavor. For details, see Table 5-28 .
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 name of the modified ECS.
progress	Integer	Reserved
links	Array of Object	Specifies ECS shortcut links. For details, see Table 5-29 .
id	String	Specifies the unique ID of an ECS.
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 field is supported in microversions later than 2.9.
description	String	Describes the ECS. This field is supported in microversions later than 2.19.

Parameter	Type	Description
tags	Array of strings	<p>Specifies ECS tags.</p> <p>This field is supported in microversions later than 2.26. If the microversion is not used for query, the response does not contain the tags field.</p> <p>Tag functions have been upgraded on the public cloud. After the upgrade, the tag values returned by the system comply with the following rules:</p> <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. • 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	<p>Specifies the ECS status.</p> <p>Options: ACTIVE, BUILD, ERROR, HARD_REBOOT, MIGRATING, REBOOT, RESIZE, REVERT_RESIZE, SHELVED, SHELVED_OFFLOADED, SHUTOFF, UNKNOWN, and VERIFY_RESIZE</p> <p>For details, see A.1 ECS Statuses.</p>

Table 5-26 addresses field description

Parameter	Type	Description
Name the network where the ECS accesses	Object	<p>Specifies the network information of the ECS.</p> <ul style="list-style-type: none"> • key indicates the network name, for example, demo_net. • value indicates the detailed network information. <p>For details, see Table 5-27.</p>

Table 5-27 Data structure of the network which an ECS accesses

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	<p>Specifies the type of an IP address. The value of this parameter can be 4 or 6.</p> <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 5-28 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID. This field is not supported in microversions later than 2.47.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 5-29 . This field is not supported in microversions later than 2.47.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor. This field is supported in microversions later than 2.47.
ram	Integer	Specifies the memory size (MB) in the ECS flavor. This field is supported in microversions later than 2.47.
disk	Integer	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. This field is supported in microversions later than 2.47.
ephemeral	Integer	Reserved This field is supported in microversions later than 2.47.
swap	Integer	Reserved This field is supported in microversions later than 2.47.
original_name	String	Specifies the name of the ECS flavor. This field is supported in microversions later than 2.47.
extra_specs	Object	Indicates an extended flavor field. For details, see os_extra_specs (flavor) Field Description . This field is supported in microversions later than 2.47.

Table 5-29 links field description

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

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.2.3 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 5-30](#) describes the parameters in the URI.

Table 5-30 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.2.4 Querying ECSs

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 5-31](#) describes the parameters in the URI.

Table 5-31 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

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

Table 5-32 Request 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.
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.
name	No	String	Specifies the ECS name, which is fuzzy matched.

Parameter	Mandatory	Type	Description
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 404. For details, see A.1 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 1000 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. The value is the tag key. NOTE Tag functions have been upgraded on the public cloud. If the tags added before the function upgrade are in the format of "Key.Value", query tags using "Key". For example, an existing tag is a.b . After the tag function upgrade, query the tag using "not-tags=a".
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.

Parameter	Mandatory	Type	Description
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.

Response

[Table 5-33](#) describes the response parameters.

Table 5-33 Response parameters

Parameter	Type	Description
servers	Array of objects	Specifies the ECSs to be queried. For details, see Table 5-34 .
servers_links	Array of objects	Specifies the link of the next page in pagination query. For details, see Table 5-35 .

Table 5-34 servers field description

Parameter	Type	Description
name	String	Specifies the ECS name.
id	String	Specifies an ECS uniquely.
links	Array of objects	Specifies ECS shortcut links. For details, see Table 5-35 .

Table 5-35 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

```
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 [9.1 Returned Values for General Requests](#).

5.2.5 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}&not-tags={not-tags}&reservation_id={reservation_id}&ip={ip}
```

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

Table 5-36 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 5-37](#) describes the request parameters.

Table 5-37 Request 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.
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 404. For details, see A.1 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 1000 ECSs are returned. For large volumes of data, you are advised to set the value to 100 .

Parameter	Mandatory	Type	Description
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. The value is the tag key. NOTE Tag functions have been upgraded on the public cloud. If the tags added before the function upgrade are in the format of "Key.Value", query tags using "Key". For example, an existing tag is a.b . After the tag function upgrade, query the tag using "not-tags=a".
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.

Response

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

Table 5-38 Response parameters

Parameter	Type	Description
servers	Array of objects	Specifies the ECSs to be queried. For details, see Table 5-39 .
servers_links	Array of objects	Specifies the link of the next page in pagination query. For details, see Table 5-41 .

Table 5-39 servers field description

Parameter	Type	Description
name	String	Specifies the ECS name.
id	String	Specifies an ECS uniquely.
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 A.1 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 time when the ECS was updated last time. The time is in the format of "2019-05-22T07:48:53Z".
flavor	Object	Specifies the ECS flavor. For details, see Table 5-40 .
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 5-46 .
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. <ul style="list-style-type: none">• key indicates the network name, for example, demo_net.• value indicates the detailed network information. For details, see Table 5-42 .

Parameter	Type	Description
security_groups	Array of objects	Specifies the security groups to which the ECS belongs. For details, see Table 5-44 .
links	Array of objects	Specifies ECS shortcut links. For details, see Table 5-41 .
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 A.1 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 A.1 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 5-43 .
fault	Object	Describes ECS faults. This parameter is optional. It is returned when an error occurs on an ECS. For details, see Table 5-45 .
description	String	Describes the ECS. This field is supported in microversions later than 2.19.
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 field is supported in microversions later than 2.16.
OS-EXT-SRV-ATTR:hostname	String	Specifies the name of the host accommodating the ECS. This field is supported in microversions later than 2.3.

Parameter	Type	Description
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the reserved ECS ID if multiple ECSs are created in a batch. This field is supported in microversions later than 2.3.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the sequence in which ECSs created in a batch start. This field is supported in microversions later than 2.3.
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 field is supported in microversions later than 2.3.
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 field is supported in microversions later than 2.3.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the ECS system disk. This field is supported in microversions later than 2.3.
OS-EXT-SRV-ATTR:user_data	String	Specifies the user data specified during ECS creation. This field is supported in microversions later than 2.3.
tags	Array of strings	Specifies ECS tags. In microversion 2.26, if the microversion is not used for query, the response does not contain the tags field. Tag functions have been upgraded on the public cloud. After the upgrade, the tag values returned by the system comply with the following rules: <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.
locked	Boolean	Specifies the ECS lock status, which is True when the ECS is locked and False when the ECS is unlocked. This field is supported in microversions later than 2.9.
accessIPv4	String	Reserved
accessIPv6	String	Reserved

Parameter	Type	Description
config_drive	String	Reserved
progress	Integer	Reserved

Table 5-40 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID. This field is not supported in microversions later than 2.47.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 5-41 . This field is not supported in microversions later than 2.47.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor. This field is supported in microversions later than 2.47.
ram	Integer	Specifies the memory size (MB) in the ECS flavor. This field is supported in microversions later than 2.47.
disk	Integer	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. This field is supported in microversions later than 2.47.
ephemeral	Integer	Reserved This field is supported in microversions later than 2.47.
swap	Integer	Reserved This field is supported in microversions later than 2.47.
original_name	String	Specifies the name of the ECS flavor. This field is supported in microversions later than 2.47.
extra_specs	Object	Extended flavor field For details, see os_extra_specs (flavor) Field Description , which is supported in microversions later than 2.47.

Table 5-41 servers_links and links field description

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

Table 5-42 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 5-43 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 field is supported in microversions later than 2.3.

Table 5-44 security_groups field description

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

Table 5-45 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 5-46 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 5-41 .

Example Request

```
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": "1",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/flavors/1",
            "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 [9.1 Returned Values for General Requests](#).

5.2.6 Querying Details About ECSs

Function

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

URI

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

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

Table 5-47 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-48](#) describes the response parameters.

Table 5-48 Response parameters

Parameter	Type	Description
server	Object	Specifies ECS information. For details, see Table 5-49 .

Table 5-49 server field description

Parameter	Type	Description
name	String	Specifies the ECS name.
id	String	Specifies an ECS uniquely.
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 A.1 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 time when the ECS was updated last time. The time is in the format of "2019-05-22T07:48:19Z".
flavor	Object	Specifies the ECS flavor. For details, see Table 5-50 .
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 5-51 .
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. <ul style="list-style-type: none"> • key indicates the network name, for example, demo_net. • value indicates the detailed network information. For details, see Table 5-53 .
security_groups	Array of objects	Specifies the security groups to which the ECS belongs. For details, see Table 5-55 .
links	Array of objects	Specifies ECS shortcut links. For details, see Table 5-52 .
tags	Array of strings	Specifies ECS tags. In microversion 2.26, if the microversion is not used for query, the response does not contain the tags field. Tag functions have been upgraded on the public cloud. After the upgrade, the tag values returned by the system comply with the following rules: <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.
os:scheduler_hints	Object	Specifies the ECS scheduling information. For details, see Table 5-57 . This parameter is not available in BMS scenarios 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. 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.

Parameter	Type	Description
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 A.1 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 A.1 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 5-54 .
fault	Object	Describes ECS faults. This parameter is optional. It is returned when an error occurs on an ECS. For details, see Table 5-56 .

Parameter	Type	Description
description	String	Describes the ECS. This field is supported in microversions later than 2.19.
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 field is supported in microversions later than 2.16.
OS-EXT-SRV-ATTR:hostname	String	Specifies the name of the host accommodating the ECS. This field is supported in microversions later than 2.3.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the reserved ECS ID if multiple ECSs are created in a batch. This field is supported in microversions later than 2.3.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the sequence in which ECSs created in a batch start. This field is supported in microversions later than 2.3.
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 field is supported in microversions later than 2.3.
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 field is supported in microversions later than 2.3.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the ECS system disk. This field is supported in microversions later than 2.3.

Parameter	Type	Description
OS-EXT-SRV-ATTR:user_data	String	Specifies the user data specified during ECS creation. This field is supported in microversions later than 2.3.
locked	Boolean	Specifies the ECS lock status, which is True when the ECS is locked and False when the ECS is unlocked. This field is supported in microversions later than 2.9.
accessIPv4	String	Reserved
accessIPv6	String	Reserved
config_drive	String	Reserved
progress	Integer	Reserved

Table 5-50 flavor field description

Parameter	Type	Description
id	String	Specifies the ECS ID. This field is not supported in microversions later than 2.47.
links	Array of objects	Specifies shortcut links for ECS types. For details, see Table 5-52 . This field is not supported in microversions later than 2.47.
vcpus	Integer	Specifies the number of vCPUs in the ECS flavor. This field is supported in microversions later than 2.47.
ram	Integer	Specifies the memory size (MB) in the ECS flavor. This field is supported in microversions later than 2.47.
disk	Integer	Specifies the system disk size in the ECS flavor. Value 0 indicates that the disk size is not limited. This field is supported in microversions later than 2.47.
ephemeral	Integer	Reserved This field is supported in microversions later than 2.47.

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

Table 5-51 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 5-52 .

Table 5-52 links field description

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

Table 5-53 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.

Parameter	Type	Description
OS-EXT-IPS:type	String	Specifies the IP address assignment mode. This is an extended attribute.

Table 5-54 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 field is supported in microversions later than 2.3.

Table 5-55 security_groups field description

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

Table 5-56 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 5-57 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 .

Parameter	Mandatory	Type	Description
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

```
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": "1",
      "links": [
        {
          "href": "http://openstack.example.com/openstack/flavors/1",
          "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 [9.1 Returned Values for General Requests](#).

5.3 Status Management

5.3.1 Starting an ECS

Function

This API is used to start a single ECS.

URI

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

[Table 5-58](#) describes the parameters in the URI.

Table 5-58 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

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

Table 5-59 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

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


Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.2 Restarting an ECS

Function

This API is used to restart a single ECS.

URI

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

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

Table 5-60 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-61](#) describes the request parameters.

Table 5-61 Request parameters

Parameter	Mandatory	Type	Description
reboot	Yes	Object	Specifies the operation to restart the ECS. For details, see Table 5-62 .

Table 5-62 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

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.3 Stopping an ECS

Function

This API is used to stop a single ECS.

URI

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

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

Table 5-63 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-64](#) describes the request parameters.

Table 5-64 Request parameters

Parameter	Mandatory	Type	Description
os-stop	Yes	Object	Specifies the operation to stop the ECS. For details, see Table 5-65 .

Table 5-65 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

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.4 Locking an ECS

Function

This API is used to lock an ECS.

You are only allowed to lock your own ECSs. After your ECSs are locked, you are not allowed to manage the ECSs.

URI

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

[Table 5-66](#) describes the parameters in the URI.

Table 5-66 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

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

Table 5-67 Request parameters

Parameter	Type	Mandatory	Description
lock	Null	Yes	Locks an ECS.

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.5 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 5-68](#) describes the parameters in the URI.

Table 5-68 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-69](#) describes the request parameters.

Table 5-69 Request parameters

Parameter	Mandatory	Type	Description
unlock	Yes	Null	Unlocks an ECS.

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.6 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 public 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 5-70 describes the parameters in the URI.

Table 5-70 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

Table 5-71 describes the request parameters.

Table 5-71 Request parameters

Parameter	Mandatory	Type	Description
createImage	Yes	Object	Creates an image using an ECS.

Table 5-72 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 field is not supported in microversions later than 2.44.
image_id	Yes	String	Specifies the image UUID. This field is supported in microversion 2.45.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.3.7 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/{project_id}/servers/{server_id}/action) or the API for rolling back ECS specifications modification (POST /v2/{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**.

To view application examples about ECS specifications modification, see [6.3 Example 3: Modifying ECS Specifications](#).

URI

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

[Table 5-73](#) describes the parameters in the URI.

Table 5-73 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

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

Table 5-74 Request parameters

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

Response

None

Example Request

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/action
{
  "resize" : {
    "flavorRef" : "s6.medium.2",
    "dedicated_host_id": "459a2b9d-804a-4745-ab19-a113bb1b4ddc"
  }
}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.8 Confirming ECS Specifications Modification

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

status=VERIFY_RESIZE

URI

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

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

Table 5-75 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-76](#) describes the request parameters.

Table 5-76 Request parameters

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

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.3.9 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

status=VERIFY_RESIZE

URI

POST /v2.1/{project_id}/servers/{server_id}/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 9.4 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
revertResize	Yes	Null	Confirms the rollback of the ECS specification modification.

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.4 Network Management

5.4.1 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 5-79](#) describes the parameters in the URI.

Table 5-79 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

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

Table 5-80 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.

Parameter	Mandatory	Type	Description
bridge	No	String	The value is fixed to be null and is in UUID format.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.4.2 Querying the Networks of a Specified ECS

Function

This API is used to query the networks of an ECS.

Constraints

None

URI

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

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

Table 5-81 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-82](#) describes the response parameters.

Table 5-82 Response parameters

Parameter	Mandatory	Type	Description
addresses	Yes	Object	Specifies the ECS network information.

Table 5-83 addresses parameter structure description

Parameter	Mandatory	Type	Description
Name of the network where the ECS accesses	Yes	Array of objects	Specifies the network where the ECS accesses. For details about the network, see Table 5-84 .

Table 5-84 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

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

Example Response

```
{
  "addresses": {
    "demo_net": [
      {
        "version": 4,
        "addr": "10.0.0.4"
      },
      {
        "version": 4,
        "addr": "192.150.73.132"
      }
    ],
    "private_net": [
      {
        "version": 4,
        "addr": "10.176.42.16"
      },
      {
        "version": 6,
        "addr": "::babe:10.176.42.16"
      }
    ]
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.4.3 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 5-85](#) describes the parameters in the URI.

Table 5-85 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Table 5-86 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 5-87](#) describes the response parameters.

Table 5-87 Response parameters

Parameter	Type	Description
Name of the network where the ECS accesses	List(Dict)	Specifies the network where the ECS accesses. For details about the network, see Table 5-88 .

Table 5-88 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

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

Example Response

```
{
  "demo_net": [
    {
      "version": 4,
      "addr": "10.0.0.4"
    },
    {
      "version": 4,
```



```
    "addr": "192.150.73.132"  
  }  
]  
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.5 Security Group Management

5.5.1 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 5-89](#) describes the parameters in the URI.

Table 5-89 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-90](#) describes the request parameters.

Table 5-90 Request parameter

Parameter	Mandatory	Type	Description
addSecurity Group	Yes	Object	Adds an ECS to a security group.

Table 5-91 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

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.5.2 Deleting a Security Group

Function

This API is used to delete a security group for an ECS.

URI

POST /v2.1/{project_id}/servers/{server_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 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

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

Table 5-93 Request parameter

Parameter	Mandatory	Type	Description
removeSecurityGroup	Yes	Object	Deletes a security group for an ECS.

Table 5-94 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

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.5.3 Querying Security Groups for a Specified ECS

Function

This API is used to query security groups for a specified ECS.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-security-groups

[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 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

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

Table 5-96 Response parameters

Parameter	Mandatory	Type	Description
security_group_s	Yes	Array of objects	Specifies security groups. For details, see Table 5-97 .

Table 5-97 security_group objects

Parameter	Mandatory	Type	Description
description	Yes	String	Specifies information about a security group. It is a string of 0 to 255 characters.
id	Yes	String	Specifies the security group ID in UUID format.
name	Yes	String	Specifies the security group name. It is a string of 0 to 255 characters.
rules	Yes	Array of objects	Specifies security group rules. For details, see Table 5-98 .
tenant_id	Yes	String	Specifies the tenant or project ID.

Table 5-98 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 5-99 . 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 5-100 . The value of ip_range or group must be empty.
id	Yes	String	Specifies the security group rule ID in UUID format.

Table 5-99 ip_range objects

Parameter	Mandatory	Type	Description
cidr	Yes	String	Specifies the peer IP segment in CIDR format.

Table 5-100 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

```
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"
        }
      ],
      "tenant_id": "bb1118612ba64af3a6ea63a1bdcaa5ae",
      "description": "default",
      "id": "3330120d-bbd1-4a73-bda9-0196a84d5670",
      "name": "default"
    }
  ]
}
```

```
    "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 [9.1 Returned Values for General Requests](#).

5.6 Flavor Management

5.6.1 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 5-101](#) describes the parameters in the URI.

Table 5-101 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

NOTE

Pagination query is supported. For details, see [1.7 Querying Data in Pages](#).

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 5-102](#) describes the query parameters.

Table 5-102 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 5-104](#) describes the response parameters.

Table 5-103 Response parameters

Parameter	Type	Description
flavors	Array of objects	Specifies ECS flavors. For details, see Table 5-104 .
flavors_links	Array of objects	Specifies data links for querying the next pages in pagination query. For details, see Table 5-105 .

Table 5-104 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 5-105 .

Parameter	Type	Description
name	String	Specifies the flavor name.

Table 5-105 links field description

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

Example Request

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors
```

Example Response

```
{
  "flavors": [
    {
      "id": "c2.medium",
      "links": [
        {
          "href": "https://compute.region.xxx.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c2.medium",
          "rel": "self"
        },
        {
          "href": "https://compute.region.xxx.com/743b4c0428d94531b9f2add666642e6b/flavors/c2.medium",
          "rel": "bookmark"
        }
      ],
      "name": "c2.medium"
    },
    {
      "id": "c2.xlarge",
      "links": [
        {
          "href": "https://compute.region.xxx.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c2.xlarge",
          "rel": "self"
        },
        {
          "href": "https://compute.region.x.com/743b4c0428d94531b9f2add666642e6b/flavors/c2.xlarge",
          "rel": "bookmark"
        }
      ],
      "name": "c2.xlarge"
    }
  ]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.6.2 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 5-106](#) describes the parameters in the URI.

Table 5-106 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

NOTE

Pagination query is supported. For details, see [1.7 Querying Data in Pages](#).

Table 5-107 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.
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. Options: asc and desc

Request

None

Response

[Table 5-108](#) describes the response parameters.

Table 5-108 Response parameters

Parameter	Type	Description
flavors	Array of objects	Specifies ECS flavors. For details, see Table 5-109 .
flavors_links	Array of objects	Specifies data links for querying the next pages in pagination query. For details, see Table 5-110 .

Table 5-109 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 .
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 .

Parameter	Type	Description
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 5-110 .

Table 5-110 links field description

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

Example Request

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/detail
```

Example Response

```
{
  "flavors": [
    {
      "name": "c3.2xlarge.2",
      "links": [
        {
          "href": "https://compute.region.xxx.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",
          "rel": "self"
        },
        {
          "href": "https://compute.region.xxx.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.xxx.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/
c3.2xlarge.4",
        "rel": "self"
      },
      {
        "href": "https://compute.region.xxx.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 [9.1 Returned Values for General Requests](#).

5.6.3 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 5-111](#) describes the parameters in the URI.

Table 5-111 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
flavors_id	Yes	Specifies the flavor ID.

Request

None

Response

[Table 5-112](#) describes the response parameters.

Table 5-112 Response parameters

Parameter	Type	Description
flavor	Object	Specifies the ECS flavor. For details, see Table 5-113 .

Table 5-113 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 "".
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 .

Parameter	Type	Description
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 5-114 .

Table 5-114 links field description

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

Example Request

```
GET https://{endpoint}/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2
```

Example Response

```
{
  "flavor": {
    "name": "c3.2xlarge.2",
    "links": [
      {
        "href": "https://compute.region.xxx.com/v2.1/743b4c0428d94531b9f2add666642e6b/flavors/c3.2xlarge.2",
        "rel": "self"
      },
      {
        "href": "https://compute.region.xxx.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 [9.1 Returned Values for General Requests](#).

5.6.4 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 5-115](#) describes the parameters in the URI.

Table 5-115 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
flavors_id	Yes	Specifies the flavor ID.

Request

None

Response

[Table 5-116](#) describes the response parameters.

Table 5-116 Response parameters

Parameter	Type	Description
extra_specs	Object	Specifies the key-value pair of an ECS flavor. For details about the returned fields, see the os_extra_specs field description in Table 4-92 .

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.7 NIC Management

5.7.1 Querying ECS NICs

Function

This API is used to query information about ECS NICs.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-interface

[Table 5-117](#) describes the parameters in the URI.

Table 5-117 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-118](#) describes the response parameters.

Table 5-118 Response parameters

Parameter	Type	Description
interfaceAttachments	Array of objects	Specifies ECS NICs. For details, see Table 5-119 .

Table 5-119 interfaceAttachments field description

Parameter	Type	Description
port_state	String	Specifies the NIC port status.

Parameter	Type	Description
fixed_ips	Array of objects	Specifies private IP addresses for NICs. For details, see Table 5-120 .
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 5-120 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

```
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 [9.1 Returned Values for General Requests](#).

5.7.2 Querying NICs of an ECS

Function

This API is used to query NICs of an ECS based on the NIC ID.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-interface/{id}

[Table 5-121](#) describes the parameters in the URI.

Table 5-121 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
id	Yes	Specifies the port ID of the NIC.

Request

None

Response

[Table 5-122](#) describes the response parameters.

Table 5-122 Response parameters

Parameter	Type	Description
interfaceAttachment	Object	Specifies ECS NICs. For details, see Table 5-123 .

Table 5-123 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 5-124 .
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 5-124 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

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface/{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 [9.1 Returned Values for General Requests](#).

5.7.3 Adding a NIC to an ECS

Function

This API is used to add a NIC to an ECS.

To view application examples about ECS NIC applications, see [6.5 Example 5: Attaching a NIC to an ECS](#).

URI

```
POST /v2.1/{project_id}/servers/{server_id}/os-interface
```

[Table 5-125](#) describes the parameters in the URI.

Table 5-125 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-126](#) describes the request parameters.

Table 5-126 Request parameters

Parameter	Mandatory	Type	Description
interfaceAttachment	Yes	Object	Specifies the NICs to be added. For details, see Table 5-127 .

Table 5-127 interfaceAttachment field description

Parameter	Mandatory	Type	Description
port_id	No	String	Specifies the port ID. Either port_id or net_id is used each time.
net_id	No	String	Specifies the network ID. Either port_id or net_id is used each time.
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 5-128 .

Table 5-128 fixed_ips field description

Parameter	Mandatory	Type	Description
ip_address	No	String	Specifies the IP address.

Response

[Table 5-129](#) describes the response parameters.

Table 5-129 Response parameters

Parameter	Type	Description
interfaceAttachment	Object	Specifies ECS NICs. For details, see Table 5-130 .

Table 5-130 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 5-131 .
port_id	String	Specifies the port ID.
net_id	String	Specifies the network ID.
mac_addr	String	Specifies the MAC address.

Table 5-131 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

```
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"
  }
}
{
  "interfaceAttachment" : {
    "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 [9.1 Returned Values for General Requests](#).

5.7.4 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/{id}

[Table 5-132](#) describes the parameters in the URI.

Table 5-132 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
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 https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-interface/{id}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.8 Disk Management

5.8.1 Querying Disks Attached to an ECS

Function

This API is used to query the disks attached to an ECS.

URI

```
GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments
```

[Table 5-133](#) describes the parameters in the URI.

Table 5-133 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

Response parameters

[Table 5-134](#) describes the response parameters.

Table 5-134 Response parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies the disks attached to an ECS. For details, see Table 5-135 .

Table 5-135 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

```
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 [9.1 Returned Values for General Requests](#).

5.8.2 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 5-136](#) describes the parameters in the URI.

Table 5-136 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
volume_id	Yes	Specifies the disk ID.

Request

None

Response

[Table 5-137](#) describes the response parameters.

Table 5-137 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disks attached to an ECS. For details, see Table 5-138 .

Table 5-138 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

```
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 [9.1 Returned Values for General Requests](#).

5.8.3 Attaching a Disk to an ECS

Function

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

To view application examples about ECS disk attachment, see [6.4 Example 4: Attaching 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 5-139](#) describes the parameters in the URI.

Table 5-139 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-140](#) describes the request parameters.

Table 5-140 Request parameters

Parameter	Mandatory	Type	Description
volumeAttachment	Yes	Object	Specifies the volumes to be attached. For details, see Table 5-141 .

Table 5-141 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 5-142](#) describes the response parameters.

Table 5-142 Response parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disks attached to an ECS. For details, see Table 5-143 .

Table 5-143 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

```
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 [9.1 Returned Values for General Requests](#).

5.8.4 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 VMTools.

When an ECS is in the **active** state, 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 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 [Formats and OSs Supported for External Image Files](#).
 - [Table 5-144](#) lists the second part of supported OSs.

Table 5-144 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	Windows Server 2008 R2 Enterprise 64bit
	Windows Server 2012 R2 Standard 64bit
	Windows Server 2016 R2 Standard 64bit
Red Hat Linux Enterprise	7.3 64bit
	6.8 64bit

6. 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.
7. Disks which are forcibly detached online will use the disk drives and PCI addresses. Therefore, the disk drives and PCI addresses will not be assigned again.
8. After a disk is forcibly detached, it still occupies the disk quota of the ECS.
9. The system disk cannot be detached forcibly online.
10. 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.
11. If logical partitions are created on the disk which is detached forcibly online, the logical partitions will become invalid.
12. 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 5-145 describes the parameters in the URI.

Table 5-145 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
volume_id	Yes	Specifies the volume ID.

Usage: DELETE /v2/{project_id}/servers/{server_id}/os-volume_attachments/
{volume_id}?delete_flag=1

Table 5-146 describes the query parameters.

Table 5-146 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

```
DELETE https://{endpoint}/v2.1/6fbc9263116a4b68818cf1edce16bc4f/servers/ab258e25-e351-47c7-b6e3-0749c5d9ed6a/os-volume_attachments/54667652-3029-4af8-9222-2d53066fd61c
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.9 Metadata Management

5.9.1 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 5-147](#) describes the parameters in the URI.

Table 5-147 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-148](#) describes the request parameters.

Table 5-148 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 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 5-149](#) describes the response parameters.

Table 5-149 Response parameters

Parameter	Type	Description
metadata	Object	Specifies the user-defined metadata key-value pair.

Example Request

```
POST https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata
{
  "metadata": {
    "key": "value"
  }
}
```

Example Response

```
{
  "metadata": {
    "key": "value"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.9.2 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 5-150](#) describes the parameters in the URI.

Table 5-150 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-151](#) describes the request parameters.

Table 5-151 Request

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 5-152](#) describes the response parameters.

Table 5-152 Response parameters

Parameter	Type	Description
metadata	Object	Specifies the user-defined metadata key-value pair.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.9.3 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 5-153](#) describes the parameters in the URI.

Table 5-153 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata/{key}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.9.4 Querying ECS Metadata

Function

This API is used to query ECS metadata.

URI

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

[Table 5-154](#) describes the parameters in the URI.

Table 5-154 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

 **NOTE**

Pagination query is not supported.

Request

None

Response

[Table 5-155](#) describes the response parameters.

Table 5-155 Response parameters

Parameter	Mandatory	Type	Description
metadata	Yes	Object	Specifies the user-defined metadata key-value pair.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/998af54b-5762-4041-abc1-f98a2c27b3a2/metadata
```

Example Response

```
{  
  "metadata": {  
    "wj": "True"  
  }  
}
```

```
}  
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.9.5 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 5-156](#) describes the parameters in the URI.

Table 5-156 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
key	Yes	Specifies the ECS metadata key.

Request

None

Response

[Table 5-157](#) describes the response parameters.

Table 5-157 Response parameters

Parameter	Type	Description
meta	Object	Specifies the user-defined metadata key-value pair.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/servers/998af54b-5762-4041-abc1-f98a2c27b3a2/metadata/key1
```

Example Response

```
{
  "meta": {
    "key1": "value1"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.9.6 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 5-158](#) describes the parameters in the URI.

Table 5-158 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
key	Yes	Specifies the ECS metadata key.

Request

[Table 5-159](#) describes the request parameters.

Table 5-159 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 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 5-160](#) describes the response parameters.

Table 5-160 Response parameters

Parameter	Type	Description
meta	Object	Specifies the user-defined metadata key-value pair.

Example Request

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/metadata/{key}
{
  "meta":{
    "key":"value"
  }
}
```

Example Response

```
{
  "meta":{
    "key":"value"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.10 Tenant Quota Management

5.10.1 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 5-161](#) describes the parameters in the URI.

Table 5-161 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 5-162](#) describes the response parameters.

Table 5-162 Response parameters

Parameter	Type	Description
limits	Object	Specifies tenant limits. For details, see Table 5-163 .

Table 5-163 limits parameter information

Parameter	Type	Description
rate	List	The value is empty.
absolute	Object	Specifies tenant quota limits. For details, see Table 5-164 .

Table 5-164 absolute parameter information

Parameter	Type	Description
maxServerMetadata	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.
maxImageMetadata	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.
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 field is not supported in microversions later than 2.35.
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.

Parameter	Type	Description
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.
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

```
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 [9.1 Returned Values for General Requests](#).

5.10.2 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 5-165](#) describes the parameters in the URI.

Table 5-165 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 5-166](#) describes the response parameters.

Table 5-166 Response parameters

Parameter	Type	Description
quota_set	Object	Specifies the quota_set object. For details, see Table 5-167 .

Table 5-167 **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.

Parameter	Type	Description
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 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 size quota of ECS groups.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.10.3 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 5-168](#) describes the parameters in the URI.

Table 5-168 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 5-169](#) describes the response parameters.

Table 5-169 Response parameters

Parameter	Type	Description
quota_set	Object	Specifies the quota_set object. For details, see Table 5-170 .

Table 5-170 quota_set parameter description

Parameter	Type	Description
cores	Integer	Specifies the quantity quota of vCPUs.

Parameter	Type	Description
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 size quota of ECS groups.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.11 Key and Password Management

5.11.1 Querying SSH Key Pairs

Function

This API is used to query SSH key pairs.

URI

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

[Table 5-171](#) describes the parameters in the URI.

Table 5-171 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 5-172](#) describes the response parameters.

Table 5-172 Response parameters

Parameter	Type	Description
keypairs	Array of objects	Specifies key pairs. For details, see Table 5-173 .

Table 5-173 keypairs field description

Parameter	Type	Description
keypair	Object	Specifies details about a key pair. For details, see Table 5-174 .

Table 5-174 keypair field description

Parameter	Type	Description
fingerprint	String	Specifies fingerprint information about the key pair.
name	String	Specifies the key pair name.
type	String	Specifies the key type, which is ssh by default. This field is supported in microversions later than 2.2.
public_key	String	Specifies information about the public key.

Example Request

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

Example Response

```
{
  "keypairs": [
    {
      "keypair": {
        "fingerprint": "15:b0:f8:b3:f9:48:63:71:cf:7b:5b:38:6d:44:2d:4a",
        "name": "keypair-601a2305-4f25-41ed-89c6-2a966fc8027a",
        "type": "ssh",
        "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC+Eo/
RZRngaGtKfs7I62ZjslIO79KklkBMXi8F+KITD4bVQHhN+kV
+4gRgkgCRbdoDqoGfpaDFs877DYX9n4z6FrAIZ4PES8TNKhatifpn9NdQYWA+IkU8CuvlEKGuFpKRi/k7JLos/
gHi2hy7QUwgtRvcefvd/vgQZOVw/mGR9Q== Generated-by-Nova\n"
      }
    }
  ]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.11.2 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 5-175](#) describes the parameters in the URI.

Table 5-175 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
keypair_name	Yes	Specifies the key pair name.

Request

None

Response

[Table 5-176](#) describes the response parameters.

Table 5-176 Response parameters

Parameter	Type	Description
keypair	Object	Specifies the SSH key pair. For details, see Table 5-177 .

Table 5-177 keypair field description

Parameter	Type	Description
public_key	String	Specifies information about the public key.
name	String	Specifies the key pair name.
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	String	Specifies the key pair ID.
updated_at	String	Specifies the time when the key pair was updated.

Parameter	Type	Description
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 field is supported in microversions later than 2.2.

Example Request

```
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:52:06:4c:c1:d3:1d:fd:06:66:01",
    "id": 1,
    "name": "keypair-3582d8b7-e588-4aad-b7f7-f4e76f0e4314",
    "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDYJrTVpcMwFqQy/
oMvtUSRofZdSRHEwrsX8AYkRvn2ZnCXm+b6+GZ2NQuuWj+ocznlnwiGFQDsL/yeE+/
kurqcPJFKKp60mToXIMyzioFw88fJtwEWawHKAclbHWpR1t4fQ4DS+/slbX/Yd9btLVQ2tpQjodGDbM9Tr9/+/
3i6rcR+EoLqmbgCgAiGiVV6VbM2Zx79yUwd+GnQejHX8BLYzoOjCnt3NREslTcmWE9FVFy6TnLmahs3FkEO/
QGgWGkaohAJlsgaVvSWGgDn2AujKYwyDokK3dXyeX3m2Vmc3ejjiqPa/C4nRrCOlko5nSgV/
9IXRx1ERlmsqZnE9usB Generated-by-Nova\n",
    "updated_at": null,
    "user_id": "fake"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.11.3 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. 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 5-178](#) describes the parameters in the URI.

Table 5-178 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 5-179](#) 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 5-179 Request parameters

Parameter	Mandatory	Type	Description
keypair	Yes	Object	Specifies the created or imported SSH key pair. For details, see Table 5-180 .

Table 5-180 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 . NOTE This field is supported in microversion 2.2.
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. NOTE This field is supported in microversion 2.10.

Response

Table 5-181 describes the response parameters.

Table 5-181 Response parameters

Parameter	Type	Description
keypair	Object	Specifies the SSH key pair. For details, see Table 5-182 .

Table 5-182 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.
private_key	String	Specifies information about the private key. <ul style="list-style-type: none"> The information about the private key is contained in the response for creating an SSH key. The information about the private key is not contained in the response for importing an SSH key.
user_id	String	Specifies the ID of the user to which the key pair belongs.
type	String	Specifies the key type. The value is ssh or x509 . NOTE This field is supported in microversion 2.2.

Example Request (Importing an SSH Key)

```
POST https://{endpoint}/v2.1/{project_id}/os-keypairs
{
  "keypair": {
    "public_key": "ssh-
rsaAAAAB3NzaC1yc2EAAAADAQABAAQDWNgtXQYeBzK9LYy4IakX7Isl5j5zqR6BU2GJaEg3RK6dLS7rKfQh
vy/V/1emK+GT/7P8up9VsMZ9Dx6PBOlow5p+2/wGsMlwDjpWiQ8zNnEMg+u/Ar/
ZhYHAMyKEAooJxIcnPoUgxfNdj/eiXV98AabsBdUA7QD30Og8F4Bmn2lii/
WD9KbQQVjb7kbB3gNIJpGTUcoX73arorqkq/ppaLRmmwMJ7bTIGl8/0MWU2Dy+eTByOaDMb2htbB
+j8ZXYeu7Oooy0NaSd+PNHv3PZ9OIVO7gd1lyoTRvCMK/
F346+zmZtk5EASSOx5RifnSwk3NtugVjXs9GMJfFLBRibGenerated-by-Nova\n\n",
    "type": "ssh",
    "name": "demo1",
    "user_id": "fake"
  }
}
```

Example Request (Creating an SSH Key)

```
POST https://{endpoint}/v2.1/{project_id}/os-keypairs
```

```
{
  "keypair": {
    "name": "demo",
  }
}
```

Example Response (Importing an SSH Key)

```
{
  "keypair": {
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQADWNgTxQYeBzK9LYy4lakX7lsl5j5zqR6BU2GJaEg3RK6dLS7rKFQhvy/
V/1emK+GT/7P8up9VsMZ9Dx6PBOLow5p+2/wGsMlwDjPWiQ8zNnEMg+u/Ar/
ZhYHAMyKEAOJxlcnPoUgxfNdj/eiXV98AabsBdUA7QD30Og8F4Bmn2lii/
WD9KbQQVjb7kbB3gNIJpGTUcoX73arorqkq/ppaLRmmwMJ7bTIGl8/0MWU2Dy+eTByOaDMb2htbB
+j8ZXyEu7Oooy0NaSd+PNHv3PZ9OIVO7gd1lyoTRvCMK/
F346+zmZtk5EASSOx5RifnSwk3NtugVjXs9GMJfFLBRib Generated-by-Nova\n\n",
    "user_id": "6fc0d2cbbfab40b199874b97097e913d",
    "name": "demo1",
    "fingerprint": "fc:47:b5:c3:7d:25:32:d5:d2:0c:19:f9:62:ac:8c:5a"
  }
}
```

Example Response (Creating an SSH Key)

```
{
  "keypair": {
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQADWNgTxQYeBzK9LYy4lakX7lsl5j5zqR6BU2GJaEg3RK6dLS7rKFQhvy/
V/1emK+GT/7P8up9VsMZ9Dx6PBOLow5p+2/wGsMlwDjPWiQ8zNnEMg+u/Ar/
ZhYHAMyKEAOJxlcnPoUgxfNdj/eiXV98AabsBdUA7QD30Og8F4Bmn2lii/
WD9KbQQVjb7kbB3gNIJpGTUcoX73arorqkq/ppaLRmmwMJ7bTIGl8/0MWU2Dy+eTByOaDMb2htbB
+j8ZXyEu7Oooy0NaSd+PNHv3PZ9OIVO7gd1lyoTRvCMK/
F346+zmZtk5EASSOx5RifnSwk3NtugVjXs9GMJfFLBRib Generated-by-Nova\n\n",
    "private_key": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpQIBAAKCAQEAljYE8UGHgcyvS2MuCGpF
+yLcJey+c6kegVnhiWhIN0SunZUu\n6yhUIb8v1f9Xpivhk/+z/
LqfVbDGFq8ejwTi6MOaftv8BrDjcAyaVokPMzZxDIPr\nvwK/2YWbWDMihADjicSHJz6FIMXzXY/
3ol1ffAGm7AXVA00A99DoPBeAZp9pYov1\ng/Sm0EFY2+5Gwd4DSCaRk1HKF
+92q6K6pKv6aWi0ZpsDCe20yBpfP9DFLNg8vknw\ncjmgzG9obWwfo/
GV8hLuzqKMTDwknfjzR79z2fTiFTu4HdZcqE0bwjCvxd+Ovs5m
\nbZORAEkjseUYn50sJNzbboFY17PRjCXxSwUYmwIDAQABAolBADNKQ+ywUA3YQLDA\nuUqLZKOB09h+0/
YccG13D5TrNaV0yaMz6h31u7pYV/RIOTXxQTXbuZt5AoR4Xca9l\nC30blmmxTDDL45CGi/T0T5AgyS7t/iuM
+smFkw12YVbv53fL7q9yCxpucdnjC95\nNj/+M3qxupiQ42uRVAYCU1jwF6J6YLy/
9UamrmVd4bWFRtT19O7uszUHLqJOZxq\n3ltqnMyD5bSMkzMN
+RxmZVXAPkBOonGVeBBInCjvHv23REkngX38zcUSc543H3Di\n4673helqSdMnl0/
TgyflQcNuOsfQcD02ABWIGBe0nCTqP8pTRo86nzK1+AoCUP72\nlsTeviECgYEA8yHKeo/
eZw25eDb3YTJovbgzA61n6AYQlDQv7rBGQDwKKQHdEqhR\nnP0PbScaoT7wSeLtYV0vxxA6qjEEuHhZlk/
t2wEILu+AH4AK88SUbUn6ZoYu+XmTA\nnx26e2QRo8Ngi/KtlfeOGXx1PM/H2/OJEN3Xjkfwjsj5bB+HjpF/
wsnUCgYEA4Yxg\nWJYJNrvSkmvXmDgxHwdxfUpVAcP40bvomNgYpKn9R2TyjMCSdIw8VVC6cGCFB9/Pc
\nG0pr8RN2SvbTaPo/96DkKdHz7NAWkzUSChD4Oy7ZNXw6GK3x1tGwMWEts1hQDHO\nnrjS
+E3bV2jC4EiVLLBxCNCbhtmQwIGUj7ZhgHM8CgYEA14UGpWpOrW8/D086LpCu
\nx46GnJmfwiRPa6dJqpfO6v9JCigv8y1i/ifR16KWP/w8HeZ1PMtgYcJd3JcaYz\nnl
+pus7JYEGxgzrPepKxN8eyDZu4nDCmnsaFfceQ02fnd2bhDhERh4oJqqRM966ax\nn+K+p0MhoF/
aqXugDF93T9kCgYEAw7TsfLFnGiJGfS4NARP11UCmUPMcf4UztX
\nIJVj7u4e9SJ6bvGfoDly3Ra8duuUtDOzDzMaSkqa4B0f//z0uEew8uCsiRVelUlx
\nZ66l1aSm8JPkTTnRmJbGDxhUXtAIVWmmy94T+AurL/IKJMFH//RdNadvPrXcuUax
\nUB5hd10CgYEA3JBuX4BriSk6Bii0kYniqFM/1tEgVelAP6DT6uePvzTFdSJ0dMQo\nnzwgWNmm43CyoKW/
rw8ylbtIQZKBFHudSNx72nSmmBKaf3QP840xsCip90ZUTfZdn\nLJzX1t4clg1wNsN4mJDwiYM9k3rB/
8EY1fh9gUYI84X6xFAHllkv0To=\n-----END RSA PRIVATE KEY-----\n",
    "user_id": "6fc0d2cbbfab40b199874b97097e913d",
    "type": "ssh",
    "name": "demo",
    "fingerprint": "fc:47:b5:c3:7d:25:32:d5:d2:0c:19:f9:62:ac:8c:5a"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.11.4 Deleting an SSH Key Pair

Function

This API is used to delete a specified SSH key pair based on the SSH key pair name.

URI

DELETE /v2.1/{project_id}/os-keypairs/{keypair_name}

[Table 5-183](#) describes the parameters in the URI.

Table 5-183 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
keypair_name	Yes	Specifies the key pair name.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/{project_id}/os-keypairs/{keypair_name}
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.11.5 Retrieving the Password for Logging In to a Windows 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.

URI

GET /v2.1/{project_id}/servers/{server_id}/os-server-password

[Table 5-184](#) describes the parameters in the URI.

Table 5-184 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-185](#) describes the response parameters.

Table 5-185 Response parameters

Parameter	Type	Description
password	String	Specifies the password in ciphertext.

Example Request

```
GET https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-server-password
```

Example Response

```
{
  "password": "UHC9+YW1xDC1Yu8Mg9n+tnOp7euEO/cW//9KgdJKWhr5w=="
}
```


Returned Values

See [9.1 Returned Values for General Requests](#).

5.11.6 Deleting the Password for Logging In to a Windows ECS

Function

This API is used to delete the recorded random password generated during initial Windows ECS installation. 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.

URI

DELETE /v2.1/{project_id}/servers/{server_id}/os-server-password

[Table 5-186](#) describes the parameters in the URI.

Table 5-186 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/{project_id}/servers/{server_id}/os-server-password
```

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.12 ECS Group Management

5.12.1 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 5-187](#) describes the parameters in the URI.

Table 5-187 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 5-188](#) describes the request parameters.

Table 5-188 Request parameters

Parameter	Mandatory	Type	Description
server_group	Yes	Object	Specifies an ECS group.

Table 5-189 server_group field description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the ECS group name. The value contains 1 to 255 characters.

Parameter	Mandatory	Type	Description
policies	Yes	Array of strings	<p>Specifies the policies associated with the ECS group. Options:</p> <ul style="list-style-type: none"> • anti-affinity: ECSs in this group must be deployed on different hosts. • affinity: ECSs in this group must be deployed on the same host. • soft-anti-affinity: ECSs in this group are deployed on different hosts if possible. If the ECSs cannot be deployed on different hosts, deploy them based on the actual condition for successful ECS creation. • soft-affinity: ECSs in this group are deployed on the same host if possible. If the ECSs cannot be deployed on the same host, deploy them based on the actual condition for successful ECS creation. <p>NOTE Only the anti-affinity policy is supported. The other three policies are retained to be compatible with native APIs. However, you are not advised to use the other three policies. You are suggested to use the policy described in 4.12.1 Creating an ECS Group.</p>

Response

[Table 5-190](#) describes the response parameters.

Table 5-190 Response parameters

Parameter	Type	Description
server_group	Object	Specifies an ECS group.

Table 5-191 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. affinity : ECSs in this group must be deployed on the same host. soft-anti-affinity : ECSs in this group are deployed on different hosts if possible. If the ECSs cannot be deployed on different hosts, deploy them based on the actual condition for successful ECS creation. soft-affinity : ECSs in this group are deployed on the same host if possible. If the ECSs cannot be deployed on the same host, deploy them based on the actual condition for successful ECS creation.
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. NOTE This parameter is added in microversion 2.13.
user_id	String	Specifies the user ID in UUID format for the ECS group. NOTE This parameter is added in microversion 2.13.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.12.2 Querying ECS Groups

Function

This API is used to query ECS groups.

URI

GET /v2.1/{project_id}/os-server-groups

[Table 5-192](#) describes the parameters in the URI.

Table 5-192 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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?

- Tenants can query only their **server_group** lists. A maximum of 1000 query results can be returned.

Request

None

Response

[Table 5-193](#) describes the response parameters.

Table 5-193 Response parameters

Parameter	Type	Description
server_groups	Array of objects	Specifies ECS groups.

Table 5-194 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.
metadata	Object	Specifies the ECS group metadata.
project_id	String	Specifies the tenant ID in UUID format for the ECS group. This field is newly added in version 2.13.
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.• affinity: ECSs in this group must be deployed on the same host.• soft-anti-affinity: ECSs in this group are deployed on different hosts if possible. If the ECSs cannot be deployed on different hosts, deploy them based on the actual condition for successful ECS creation.• soft-affinity: ECSs in this group are deployed on the same host if possible. If the ECSs cannot be deployed on the same host, deploy them based on the actual condition for successful ECS creation. NOTE Only the anti-affinity policy is supported.
user_id	String	Specifies the user ID in UUID format for the ECS group. This field is newly added in version 2.13.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.12.3 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 5-195](#) describes the parameters in the URI.

Table 5-195 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group UUID.

Request

None

Response

[Table 5-196](#) describes the response parameters.

Table 5-196 Response parameters

Parameter	Type	Description
server_group	Object	Specifies an ECS group.

Table 5-197 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.
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.
user_id	String	Specifies the user ID in UUID format for the ECS group.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.12.4 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 5-198](#) describes the parameters in the URI.

Table 5-198 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_group_id	Yes	Specifies the ECS group UUID.

Request

None

Response

None

Example Request

```
DELETE https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/os-server-groups/  
5bbcc3c4-1da2-4437-a48a-66f15b1b13f9
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.13 ECS Operation Management

5.13.1 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 5-199](#) describes the parameters in the URI.

Table 5-199 Path parameters

Parameter	Mandator y	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Parameter	Mandatory	Description
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-200](#) describes the response parameters.

Table 5-200 Response parameters

Parameter	Type	Description
instanceActions	Array of Object	Specifies operations performed on the ECS. For details, see Table 5-201 .

Table 5-201 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, suspend, resume, rescue, unrescue, changePassword, shelve, unshelve, live-migration, live_migration_cancel, live_migration_force_complete, trigger_crash_dump, and 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.

Parameter	Mandatory	Type	Description
updated_at	Yes	String	Specifies the time when the information was updated.
start_time	Yes	String	Specifies the time when the action was started.
user_id	Yes	String	Specifies the user ID.

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.13.2 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 5-202](#) describes the parameters in the URI.

Table 5-202 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
request_id	Yes	Specifies the request ID.

Request

None

Response

[Table 5-203](#) describes the response parameters.

Table 5-203 Response parameters

Parameter	Type	Description
instanceAction	Object	Specifies an operation performed on the ECS. For details, see Table 5-204 .

Table 5-204 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 5-205 .

Table 5-205 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

```
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": "752be40780484291a9cc7ae50fff3e6d",
    "start_time": "2014-12-11T02:17:49.000000",
    "request_id": "req-5a429946-c9cc-45cc-b5bd-68864209e5cc",
    "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 [9.1 Returned Values for General Requests](#).

5.14 ECS Console Management

5.14.1 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 5-206](#) describes the parameters in the URI.

Table 5-206 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 5-207](#) describes the request parameters.

Table 5-207 Request parameters

Parameter	Mandatory	Type	Description
os-getConsoleOutput	Yes	Object	Obtains ECS management console logs.

Table 5-208 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

None

Example Request

```
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 [9.1 Returned Values for General Requests](#).

5.14.2 Obtaining a VNC-based Remote Login Address (Microversion 2.6 or Later)

Function

This API is used to obtain the address for remotely logging in to an ECS using VNC.

URI

POST /v2.1/{project_id}/servers/{server_id}/remote-consoles

[Table 5-209](#) describes the parameters in the URI.

Table 5-209 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Constraints

- When using this API, ensure that the microversion is 2.6 or later.
Add a microversion using the HTTP request header X-OpenStack-Nova-API-Version or OpenStack-API-Version.
For example, X-OpenStack-Nova-API-Version: 2.6 or OpenStack-API-Version: compute 2.6
- An obtained login address is valid for 10 minutes. Obtain a new one after expiration.

Request

Table 5-210 Request parameters

Parameter	Mandatory	Type	Description
remote_console	Yes	Object	Obtains the address for remotely logging in to an ECS using VNC.

Table 5-211 remote_console parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies a remote login mode. Set it to novnc .
protocol	Yes	String	Specifies a remote login protocol. Set it to vnc .

Response

[Table 5-212](#) describes the response parameters.

Table 5-212 Response parameters

Parameter	Type	Description
remote_console	Object	Obtains the address for remotely logging in to an ECS.

Table 5-213 remote_console parameters

Parameter	Type	Description
type	String	Specifies a remote login mode.
protocol	String	Specifies a remote login protocol.
url	String	Specifies a remote login URL. The URL is valid for 10 minutes. Obtain a new one after expiration.

Example Request

```
POST https://{endpoint}/v2.1/13c67a214ced4afb88d911ae4bd5721a/servers/47bc79ae-df61-4ade-9197-283a74e5d70e/remote-consoles
```



```
{
  "remote_console": {
    "type": "novnc",
    "protocol": "vnc"
  }
}
```

Example Response

```
{
  "remote_console": {
    "url": "https://nova-novncproxy.az21.dc1.domainname.com:8002/vnc.auto.html?token=80fa7c8d-37fe-451e-8b08-bfbd9fb6a4df&lang=EN",
    "type": "novnc",
    "protocol": "vnc"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

Error Codes

See [9.2 Error Code Description](#).

5.15 AZ

5.15.1 Querying AZs

Function

This API is used to query AZs.

URI

GET /v2.1/{project_id}/os-availability-zone

[Table 5-214](#) describes the parameters in the URI.

Table 5-214 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Response

[Table 5-215](#) describes the response parameters.

Table 5-215 Response parameters

Parameter	Type	Description
availabilityZoneInfo	Array of objects	Specifies the AZ information.

Table 5-216 AvailabilityZoneInfo parameter information

Parameter	Type	Description
zoneState	Object	Specifies the AZ status.
hosts	List	The parameter is set to null .
zoneName	String	Specifies the AZ name.

Table 5-217 zoneState parameter information

Parameter	Type	Description
available	Boolean	Specifies the AZ status.

Example Request

```
GET https://{endpoint}/v2.1/9c53a566cb3443ab910cf0daebca90c4/os-availability-zone
```

Example Response

```
{
  "availabilityZoneInfo": [{
    "zoneState": {
      "available": true
    },
    "hosts": null,
    "zoneName": "az1.dc1"
  },
  {
    "zoneState": {
      "available": true
    },
    "hosts": null,
    "zoneName": "vmware.az1"
  }
}]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.16 Tag Management

5.16.1 Querying 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 5-218](#) describes the parameters in the URI.

Table 5-218 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

[Table 5-219](#) describes the response parameters.

Table 5-219 Response parameters

Parameter	Type	Description
tags	Array of strings	Specifies ECS tags.

Example Request

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

Example Response

```
Example response
{
  "tags": ["baz=xyy", "foo", "qux"]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.16.2 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 5-220](#) describes the parameters in the URI.

Table 5-220 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 5-221](#) describes the request parameters.

Table 5-221 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 5-222 Response parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of strings	Specifies ECS tags.

Table 5-223 Reserved tag parameters

Tag Name	Description
__type_bare metal	Specifies that the server is a BMS.
__type_virt ual	Specifies that the server is an ECS.

Example Request

```
PUT https://{endpoint}/v2.1/{project_id}/servers/{server_id}/tags
{
  "tags": ["baz", "foo", "qux"]
}
```

Example Response

```
{
  "tags": ["baz", "foo", "qux"]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

5.16.3 Deleting Tags from 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 5-224](#) describes the parameters in the URI.

Table 5-224 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

None

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.16.4 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 5-225](#) describes the parameters in the URI.

Table 5-225 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.
tag	Yes	Specifies the key of the tag to be added. NOTE Tag functions have been upgraded on the public cloud. If the tags added before the function upgrade are in the format of "Key.Value", query tags using "Key". For example, an existing tag is "a.b". The tag can be queried in the format of "tag=a.b" before and in the format of "tag=a" now according to the new tag rules.

Request

None

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.16.5 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 5-226](#) describes the parameters in the URI.

Table 5-226 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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. NOTE Tag functions have been upgraded on the public cloud. If the tags added before the function upgrade are in the format of "Key.Value", query tags using "Key". For example, an existing tag is "a.b". The tag can be queried in the format of "tag=a.b" before and in the format of "tag=a" now according to the new tag rules.

Request

None

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.16.6 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 5-227](#) describes the parameters in the URI.

Table 5-227 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Parameter	Mandatory	Description
tag	Yes	<p>Specifies the key of the tag to be deleted. If no key is specified, all tags of the ECS will be deleted.</p> <p>NOTE</p> <p>Tag functions have been upgraded on the public cloud. If the tags added before the function upgrade are in the format of "Key.Value", delete tags using "Key".</p> <p>For example, an existing tag is a.b. After the tag function upgrade, delete the tag using "a".</p>

Request

None

Response

None

Example Request

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

Example Response

None

Returned Values

See [9.1 Returned Values for General Requests](#).

5.17 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.

6 Application Examples

6.1 Example 1: Creating an ECS

Scenarios

This section describes how to create an ECS by calling APIs. For details, see [3 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.

Constraints

The ECS created using this API is in pay-per-use billing mode.

Involved APIs

Creating an ECS involves viewing flavors and AZs as well as creating EVS disks. The following APIs are required:

- [Querying AZs](#): Determine the AZ where the ECS to be created is located.
- [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 AZ where the ECS is located.

1. View AZs.

- API
URI format: GET /v2.1/{project_id}/os-availability-zone
For details, see [5.15.1 Querying AZs](#).
- Example request
GET: `https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/os-availability-zone`
Obtain `{endpoint}` from [Regions and Endpoints](#).
- Example response

```
{
  "availabilityZoneInfo": [
    {
      "hosts": null,
      "zoneState": {
        "available": true
      },
      "zoneName": "zone_01"
    },
    {
      "hosts": null,
      "zoneState": {
        "available": true
      },
      "zoneName": "zone_01"
    }
  ]
}
```

2. Select an AZ based on site requirements and record the AZ (**zoneName**).

Step 2 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 [5.6.2 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": 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 3 Determine the image.

1. View images.

- API

URI format: GET /v2.1/{project_id}/images/detail

For details, see [10.1.2 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 4 Determine the network configuration.

1. View networks.

- API

URI format: GET /v2.1/{project_id}/os-networks

For details, see [5.4.1 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 5 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 [5.11.3 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
5pumOV05SiO3WCswnaQ5xMdOL31mNiHtwlwq9dJi7X6jJBB2shTD
+00G5WuwkBbFU4CLvt1B44u0NUiaTJ35NAvW2/4XvpXm9OwiQ3B5ge6ZY7Esi38Unh
+pkbhPKYxNBCK8yoOlojQhWs75abdxZBi811/8RwLcNiFiocA2RGxtRjBdpEScj+1TU
+OcfZdQnr0AFbO11z7yxflygwwzVTgUuJNbMbKHStQqRbklfMIHY4RBPQgb7RN/YaXKTQsXT84k
+D9xLDNo7Wj4fwOJTOz/s/PvblOqjRHt9D6Y4IKd Generated-by-Nova\n",
    "private_key": "-----BEGIN RSA PRIVATE KEY-----
\nMII EoglBAAKCAQEAq0eRnMJYeYoeyTr8yFLkMUuas1ljzGB3g15AykkPVuabpJld
\nOUojt1grMJ2kOcTHTPd9ZjYh7cJcKvXSYu1+oyQQdriUw/tNBuVrsJAWxVOAi77d
\nQeOLtDVlImkyd+TQL1tv+F76V5vTsIkNweYHumWOxLit/FJ4fqZG4T5GMTQQivMqD\nnpal0IVrO
+Wm3cWQYvNdf/EcC3DYhYqHANKRsbUYwXaREnl/tU1PjnH2XUJ69ABWz\ntdc
+8sXyMoMMM1U4FLiTWzGyhOrUKkW5JXzJR2OEqT0IG+0Tf2Glyk0El0/OJPG/\ncZQzaO1o
+H8DiUzs/7Pz72yDqo0R7fQ+mOCCnQIDAQABAoLBAA6/c9dGmK2mae4z\nyQ5KroFdvC1TNhej
+sZx+CwyzEJUSvSuHcvQCXFBaz8FY92hhvPKcX66jINXZ
+4/\nCMWAQ5YyhRiow0Y91HvsS0bywoknX3q6kxBfodmyyCWFkgd5iMTADb1Lx0a27Y7\njLS4DI5
gyiGmxUN2Ng24wWEAjE8Znu0lrtr5IZKp+s5IAi/rb5AG/mlL7EzicE8c\nnmGP+QAa
+nzwhAwNhFwVID230xen/ZcoL1d77hxeARNqJUxoR25gwd6Ebg2y9pDW
\nVu6cbbzgdGUCfQYIMEoAamAkCswOsDpVDBXwQnt2A537n6Wq2bgYIKusHr9thtXp\n/
5ubQLUCgYEA4zYuBG2vtLHnvce26P8o2j1xclS9K0ozkah9JfL3hqFN0sAqLz7\n/
Fm1jA4kzHJS3d0UqP3AMDxY3HklqCn4Be7lqeAAe2AfqkOZpt9MDNv4VwKe9sPb
\nViW1qjL3FzILC/YWTRNSlpwRjqJGhA+UQt8rOia1k/zXmrEs7bXLCgYEAwPsu
\nK3j5QoAiziYVMYf5iCzWwAM9Ljpf9gw23leftDlzhhfFtjplVRsYxRGU0Uz84GMI\nnTd5zmcIF/
1KUfhqmeiQzz6NIPEYEReahjpQ/sOH/Gk5Rwr3QwYPrwAu5x+kk/SRI\nnKPKqw7APTR0sMQBCuq
+ZYwGYLGPmdd1zUdLfb0sCgYBkuz11iydtxb3G/obSD2WO\nnM9ValycmzRPFzNwGRH/
gOR0mhTluKp0wyjybSd34oeqph/2r2ivddrOysxoqa8jg
\n4IQDZyLvj7MaKjQxrieqP89+y9Or9TMFo1xB46x2G8EN8/xHuA9YGnZSPFtWw72m
\nhRqV0hv82amWsA0vHnRUswKBgDsKHxvrTmbNkNhkykMXCH5iyWiBFSyZa1ZJMlGf
\nknsqfdeVPwF6E55QKAN2uuTlwzG/3ljPxAhR1hvmUJQN9JSBiUKbtW6GPCRvbr\nf/
jLi1lu99COZdluVKeybqn8Z/aSNP24DR9FM8kxzZ1IMPaTBmhFypp6BchlclBt
\nxTG1AoGAfcrkVbV1SOy7fECUtmPUeCcw0yU4GWj3sR2RbII63C500RVYQlUpUaRR
\naANbASHTVR4myOKtGSxEUhAQHlxFDwsDL7W3gzAqTFbEDp1xAAYt/
nkOAhQjEm4\nnORFdDETeXLQG1KMUj+8AdnhfYp3JTdft6rmpPEBUFICAUMAvb0=\n-----END
RSA PRIVATE KEY-----\n",
    "user_id": "f79791beca3c48159ac2553fff22e166",
    "name": "demo1",
    "fingerprint": "57:a7:a2:ed:5f:aa:e7:54:62:2e:bb:e7:92:22:cb:40"
  }
}

```

2. Import the key pair.

- API

URI format: POST `/v2.1/{project_id}/os-keypairs`

For details, see [5.11.3 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
AAAAB3NzaC1yc2EAAAADAQABAAQDY8wMTdBYiJgi62o6eShoOISkx3CZ3cE6PHisDblfK3Y0B
g7EHV7iV9c74pqsrlhK0xuGUuO1NxDQWbkwLTPN4F9ly5CIYohLuMlpln6LDtfrPpdhEh3lxL8MM6
1gyfpKzeKkwkEpSFj27Rgh6zCyJgBpkA2A0HTP737UlitahL4faCWDIS
+Vj6mbcfkWiMhuMczTZgSKAZ4PfoG4B5HJhR52C6A4XLiQFT9heh9gnlSg
+uTogTKUbcjKuN7M6AraJpul6eHhV9YI4433sDmuiBF/njvreVPWwAHlAkgT9I8q1T/
cfEFiwzXpdGbkK5O8NC7K+qNbbdKihlahONT Generated-by-Nova\n",
    "type": "ssh",
    "name": "demo2",
    "user_id": "fake"
  }
}

```

- Example response

```
{
  "keypair": {
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDY8wMTdBYiJgi62o6eShoOISKx3CZ3cE6PHisDblfK3Y0B
g7EHV7iV9c74pqsrIhK0xuGUuO1NxDQWbkwLTPN4F9ly5CIYohLuMlpln6LDtfRPpdhEh3lxL8MM6
1gyfpKzeKkwkEpSFj27Rgh6zCyJgBpkA2A0HTP737UlitahL4faCWDIS
+Vj6mbcfkWiMhuMCzTZgSKAZ4PfoG4B5HJhR52C6A4XLIQFT9heh9gnlslG
+uTogTKUbcjKuN7M6AraJpul6eHhV9YI4433sDmuiBF/njvreVPWwAHLIAkgT9I8q1T/
cfEFiwzXpdGbkK5O8NC7K+qNbbdKihlahONT Generated-by-Nova\n",
    "user_id": "f79791beca3c48159ac2553fff22e166",
    "name": "demo2",
    "fingerprint": "dd:44:45:d9:f6:4f:c0:24:2d:81:aa:c4:4b:83:c2"
  }
}
```

- Record the name in the response body, for example, **demo2**.

Step 6 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 [5.2.1 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

```
{
  "server": {
    "security_groups": [
      {
        "name": "default"
      }
    ]
  }
}
```

```
],
  "OS-DCF:diskConfig": "MANUAL",
  "links": [
    {
      "rel": "self",
      "href": "https://None/v2.1/74610f3a5ad941998e91f076297ecf27/servers/6d311127-bce1-48db-bf0f-cac9f8f7f077"
    },
    {
      "rel": "bookmark",
      "href": "https://None/74610f3a5ad941998e91f076297ecf27/servers/6d311127-bce1-48db-bf0f-cac9f8f7f077"
    }
  ],
  "id": "6d311127-bce1-48db-bf0f-cac9f8f7f077",
  "adminPass": "WcC4QoVZPXpV"
}
```

Step 7 Verify the ECS creation.

- API

URI format: GET /v2.1/{project_id}/servers/{server_id}

For details, see [5.2.6 Querying Details About ECSs](#).

- 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": "ztttestvm1",
"progress": 0,
"OS-SRV-USG:launched_at": "2018-05-25T01:47:55.755922",
"updated": "2018-05-25T01:47:55Z",
"status": "ACTIVE"
}
}
```

----End

6.2 Example 2: Querying ECSs

Scenarios

This section describes how to use the API for querying details about ECSs to obtain all ECSs of a tenant by page.

The operations described in this section include information query by page and data filtering and sorting. For details about the parameters, see [5.2.5 Querying Details About ECSs](#).

Involved APIs

Querying ECSs involves the following APIs:

- [Querying Details About ECSs by Specifying the Maximum Number of ECSs Displayed on One Page](#)
- [Querying Details About ECSs by Specifying the Maximum Number of ECSs Displayed on One Page and the ID of the Last Flavor on One Page](#)

Procedure

Step 1 Query details about ECSs by specifying the maximum number of ECSs displayed on one page.

- API
URI format: GET /v2.1/{project_id}/servers/detail
For details, see [5.2.5 Querying Details About ECSs](#).
- Example request
GET: `https://{endpoint}/v2.1/743b4c0428d945316666666666666666/servers/detail?limit=100`
Obtain `{endpoint}` from [Regions and Endpoints](#).
The **limit** value can be adjusted based on ECS data.
- Example response

```
{
  "servers": [
    .....
    {
      "tenant_id": "743b4c0428d945316666666666666666",
      "metadata": {
      },
      "addresses": {
        "140fd038-c4ae-4c32-ac07-34b525eb6b95": [
          {
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:e9:91:50",
            "addr": "192.168.0.178",
            "OS-EXT-IPS:type": "fixed",
            "version": 4
          }
        ]
      },
      "OS-EXT-STTS:task_state": null,
      "OS-DCF:diskConfig": "MANUAL",
      "OS-EXT-AZ:availability_zone": "xxx",
      "links": [
        {
          "rel": "self",
          "href": "https://ecs.xxx/v2.1/743b4c0428d945316666666666666666/servers/f215afe8-b0c2-41cc-9191-585638166812"
        },
        {
          "rel": "bookmark",
          "href": "https://ecs.xxx/743b4c0428d945316666666666666666/servers/f215afe8-b0c2-41cc-9191-585638166812"
        }
      ],
      "OS-EXT-STTS:power_state": 4,
      "id": "f215afe8-b0c2-41cc-9191-585638166812",
      "os-extended-volumes:volumes_attached": [
        {
          "id": "546cf622-b9e5-4784-b659-6881e711f283"
        }
      ],
      "OS-EXT-SRV-ATTR:host": "pod01.xxx",
      "accessIPv4": "",
      "image": {

```



```
{
  "tenant_id": "743b4c0428d945316666666666666666",
  "metadata": {
  },
  "addresses": {
    "140fd038-c4ae-4c32-ac07-34b525eb6b95": [
      {
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:a5:2b:f8",
        "addr": "192.168.0.169",
        "OS-EXT-IPS:type": "fixed",
        "version": 4
      }
    ]
  },
  "OS-EXT-STS:task_state": null,
  "OS-DCF:diskConfig": "MANUAL",
  "OS-EXT-AZ:availability_zone": "xxx",
  "links": [
    {
      "rel": "self",
      "href": "https://xxx/v2.1/743b4c0428d945316666666666666666/servers/62348919-0188-43ec-aae6-82c1e96c49eb"
    },
    {
      "rel": "bookmark",
      "href": "https://ecs.xxx/743b4c0428d945316666666666666666/servers/62348919-0188-43ec-aae6-82c1e96c49eb"
    }
  ],
  "OS-EXT-STS:power_state": 4,
  "id": "62348919-0188-43ec-aae6-82c1e96c49eb",
  "os-extended-volumes:volumes_attached": [
    {
      "id": "f0bb068a-61c1-4dc8-8455-09857773c3ff"
    }
  ],
  "OS-EXT-SRV-ATTR:host": "pod01.xxx",
  "accessIPv4": "",
  "image": {
    "links": [
      {
        "rel": "bookmark",
        "href": "https://ecs.xxx/743b4c0428d945316666666666666666/images/3a64bd37-955e-40cd-ab9e-129db56bc05d"
      }
    ]
  },
  "id": "3a64bd37-955e-40cd-ab9e-129db56bc05d"
},
"OS-SRV-USG:terminated_at": null,
"accessIPv6": "",
"created": "2019-07-27T03:06:48Z",
"hostId": "31397656d6b318d01431f60c481d8425dc58eb421d237a385ceb80ee",
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova022@36",
"flavor": {
  "links": [
    {
      "rel": "bookmark",
      "href": "https://ecs.xxx/743b4c0428d945316666666666666666/flavors/s3.medium.4"
    }
  ]
},
"id": "s3.medium.4"
},
"key_name": null,
"security_groups": [
  {
    "name": "sg-1e22"
  }
],
"config_drive": "",
```

```
    "OS-EXT-STS:vm_state": "stopped",
    "user_id": "f7e10ccf7abc4757b483895c3e06964a",
    "OS-EXT-SRV-ATTR:instance_name": "instance-004a0eea",
    "name": "test-dx",
    "OS-SRV-USG:launched_at": "2019-07-27T03:07:05.000000",
    "updated": "2019-08-13T03:12:38Z",
    "status": "SHUTOFF"
  }
],
"servers_links": [
  {
    "rel": "next",
    "href": "https://ecs.xxx/v2.1/743b4c0428d945316666666666666666/servers/detail?
limit=100&marker=62348919-0188-43ec-aae6-82c1e96c49eb"
  }
]
}
```

Step 3 Collect query results.

Repeat step [Step 1](#) until the returned query result is empty or the returned body does not contain the `servers_links` field. This indicates that all ECSs have been queried.

The collected ECSs are the desired query results.

----End

6.3 Example 3: Modifying ECS Specifications

Scenarios

When ECS specifications fail to meet service requirements, they can be modified, for example, by upgrading the vCPUs and memory. Certain ECSs also support changing ECS types during specifications modification.

Constraints

- You can modify the ECS specifications only when the ECS is stopped.
- The EVS disk capacity of the ECS cannot be reduced during the specifications modification.
- When modifying the specifications of an ECS, you are not allowed to select sold-out vCPU and memory resources.
- ECS specifications (vCPU or memory) degrade deteriorates the ECS performance.
- Certain ECSs do not support specifications modification. To query the target flavors to which a specified ECS flavor can be changed, use [API 4.4.3 Querying the Target Flavors to Which an ECS Flavor Can Be Changed \(Discarded\)](#).

Involved APIs

Modifying ECS specifications involves the following APIs:

- [Modifying the Specifications of an ECS](#)

- [Confirming ECS Specifications Modification](#)
- [Rolling Back ECS Specifications Modification](#)

Procedure

Step 1 Modify ECS specifications.

- API
URI format: POST /v2.1/{tenant_id}/servers/{server_id}/action
For details, see [Modifying the Specifications of an ECS](#).
- Example request

```
{
  "resize": {
    "flavorRef": "s6.medium.2"
  }
}
```
- Example response
N/A

Step 2 Confirm the specifications modification.

The ECS must be in **resized** state, **OS-EXT-STS:vm_state** being set to **resized**.

- API
URI format: POST /v2.1/{tenant_id}/servers/{server_id}/action
For details, see [5.3.8 Confirming ECS Specifications Modification](#).
- Example request

```
{
  "confirmResize": null
}
```
- Example response
N/A

Step 3 (Optional) Roll back the specifications modification.

Notes:

The ECS must be in **resized** state, **OS-EXT-STS:vm_state** being set to **resized**.

The data modified during specifications modification will be lost after the rollback.

- API
URI format: POST /v2.1/{tenant_id}/servers/{server_id}/action
For details, see [5.3.9 Rolling Back ECS Specifications Modification](#).
- Example request

```
{
  "revertResize": null
}
```
- Example response
N/A

----End

6.4 Example 4: Attaching a Disk to an ECS

Scenarios

If the existing disks of an ECS fail to meet service requirements, for example, due to insufficient disk space or poor disk performance, you can attach more available disks to the ECS, or call the EVS disk creation API to create disks and attach them to the ECS. To attach an EVS disk to an ECS, you need to call the desired API.

A data disk can be attached by setting the **data_volumes** parameter during ECS creation or after the ECS is created. This section describes how to attach a disk to a created ECS.

Involved APIs

Attaching a disk involves the following APIs:

- [Creating EVS Disks](#)
- [Attaching a Disk to an ECS](#)
- [Querying Disk Attachment of an ECS](#)

Procedure

Step 1 Create an EVS disk.

1. Create an EVS disk.

– API

URI format: POST `/v2/{tenant_id}/volumes`

For details, see [Creating EVS Disks](#).

– Example request

POST: `https://{endpoint}/v2/74610f3a5ad941998e91f076297ecf27/volumes`

Obtain `{endpoint}` from [Regions and Endpoints](#).

Body:

```
{
  "volume": {
    "name": "openapi_vol02",
    "availability_zone": "az_test_01",
    "description": "create for api test",
    "volume_type": "SSD",
    "size": 40
  }
}
```

– Example response

```
{
  "volume": {
    "status": "creating",
    "user_id": "f79791beca3c48159ac2553fff22e166",
    "attachments": [],
    "links": [
      {
        "href": "https://xxx/v2/74610f3a5ad941998e91f076297ecf27/volumes/51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",
      }
    ]
  }
}
```

```
    "rel": "self"
  },
  {
    "href": "https://xxx/74610f3a5ad941998e91f076297ecf27/volumes/51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",
    "rel": "bookmark"
  }
],
"availability_zone": "az_test_01",
"bootable": "false",
"encrypted": false,
"created_at": "2018-05-16T11:19:33.992984",
"description": "create for api test",
"updated_at": null,
"volume_type": "SSD",
"name": "openapi_vol02",
"replication_status": "disabled",
"consistencygroup_id": null,
"source_volid": null,
"snapshot_id": null,
"shareable": false,
"multiattach": false,
"metadata": {
  "__system__volume_name": "openapi_vol02"
},
"id": "51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",
"size": 40
}
}
```

- Record the **volume** ID in the response.

Step 2 Attach the disk to the ECS.

- API

URI format: POST /v2.1/{tenant_id}/servers/{server_id}/os-volume_attachments

For details, see [5.8.3 Attaching a Disk to an ECS](#).

- Example request

`https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers/9f4d9281-95e7-4915-a126-1ee597101e2e/os-volume_attachments`

Obtain `{endpoint}` from [Regions and Endpoints](#).

Body:

```
{
  "volumeAttachment": {
    "volumelId": "51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",
    "device": "/dev/sdb"
  }
}
```

- Example response

```
{
  "volumeAttachment": {
    "id": "51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",
    "volumelId": "51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",
    "serverId": "9f4d9281-95e7-4915-a126-1ee597101e2e",
    "device": "/dev/sdb"
  }
}
```

Step 3 Verify the disk attachment.

- API

URI format: GET /v2.1/{tenant_id}/servers/{server_id}/os-volume_attachments

For details, see [Querying Disks Attached to an ECS](#).

- Example request

```
https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers/  
9f4d9281-95e7-4915-a126-1ee597101e2e/os-volume_attachments
```

Obtain *{endpoint}* from [Regions and Endpoints](#).

- Example response

```
{  
  "volumeAttachments": [  
    {  
      "volumeId": "4fc0b4cc-9d6c-431c-be70-3dfeec2ff6e0",  
      "id": "4fc0b4cc-9d6c-431c-be70-3dfeec2ff6e0",  
      "device": "/dev/sda",  
      "serverId": "9f4d9281-95e7-4915-a126-1ee597101e2e"  
    },  
    {  
      "volumeId": "51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",  
      "id": "51f45e08-1d4f-44c6-a4a9-84a488e0e8d3",  
      "device": "/dev/sdb",  
      "serverId": "9f4d9281-95e7-4915-a126-1ee597101e2e"  
    }  
  ]  
}
```

----End

6.5 Example 5: Attaching a NIC to an ECS

Scenarios

If an ECS requires multiple NICs, you can call the API for creating NICs and attach them to the ECS.

A NIC can be attached by setting the **nics** parameter during ECS creation or after the ECS is created. This section describes how to attach a NIC to a created ECS.

Involved APIs

Attaching a NIC involves the following APIs:

- [Creating a Network](#)
- [Creating a Subnet](#)
- [Creating a Port](#)
- [Adding a NIC to an ECS](#)
- [Query NICs of an ECS](#)

Procedure

Step 1 Create a NIC.

1. Create a network.
 - API
URI format: POST /v1/{project_id}/vpcs
For details, see [Creating a VPC](#).

- Example request

POST `https://{Endpoint}/v1/{project_id}/vpcs`

Obtain *{endpoint}* from [Regions and Endpoints](#).

Body:

```
{
  "vpc": {
    "name": "vpc",
    "description": "test",
    "cidr": "192.168.0.0/16"
  }
}
```

- Example response

```
{
  "vpc": {
    "id": "99d9d709-8478-4b46-9f3f-2206b1023fd3",
    "name": "vpc",
    "description": "test",
    "cidr": "192.168.0.0/16",
    "status": "CREATING",
    "routes": []
  }
}
```

2. Record the **vpc** ID in the response.

3. Create a subnet.

- API

URI format: POST `/v2.0/subnets`

For details, see [Creating a Subnet](#).

- Example request

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

Obtain *{endpoint}* from [Regions and Endpoints](#).

Body:

```
{
  "subnet": {
    "name": "testsubnet",
    "enable_dhcp": true,
    "network_id": "c4a3019d-1ac0-4cfb-a838-2342eb992e6b",
    "tenant_id": "74610f3a5ad941998e91f076297ecf27",
    "dns_nameservers": [
      "8.8.8.8",
      "8.8.8.7"
    ],
    "allocation_pools": [
      {
        "start": "10.0.10.2",
        "end": "10.0.10.254"
      }
    ],
    "host_routes": [],
    "ip_version": 4,
    "gateway_ip": "10.0.10.1",
    "cidr": "10.0.10.0/24"
  }
}
```

- Example response

```
{
  "subnet": {
    "name": "testsubnet",
    "cidr": "10.0.10.0/24",
    "id": "877b5567-e8c6-4a0d-aabf-0f13da225fe5",
  }
}
```

```
"enable_dhcp": true,
"network_id": "c4a3019d-1ac0-4cfb-a838-2342eb992e6b",
"tenant_id": "74610f3a5ad941998e91f076297ecf27",
"dns_nameservers": [
  "8.8.8.8",
  "8.8.8.7"
],
"allocation_pools": [
  {
    "start": "10.0.10.2",
    "end": "10.0.10.254"
  }
],
"host_routes": [],
"ip_version": 4,
"gateway_ip": "10.0.10.1"
}
}
```

4. Record the **subnet** ID in the response.

5. Create a port.

– API

URI format: POST /v2.0/ports

For details, see [Creating a Port](#).

– Example request

POST: <https://{endpoint}/v2.0/ports>

Obtain *{endpoint}* from [Regions and Endpoints](#).

Body:

```
{
  "port": {
    "admin_state_up": true,

    "fixed_ips": [
      {
        "subnet_id": "877b5567-e8c6-4a0d-aabf-0f13da225fe5"
      }
    ],
    "name": "test",
    "network_id": "c4a3019d-1ac0-4cfb-a838-2342eb992e6b",
    "tenant_id": "74610f3a5ad941998e91f076297ecf27"
  }
}
```

– Example response

```
{
  "port": {
    "id": "7bf1c36f-e7f8-478a-be3d-674b486abbc4",
    "name": "test",
    "status": "DOWN",
    "admin_state_up": true,
    "fixed_ips": [
      {
        "subnet_id": "877b5567-e8c6-4a0d-aabf-0f13da225fe5",
        "ip_address": "10.0.10.233"
      }
    ],
    "mac_address": "fa:16:3e:db:91:f6",
    "network_id": "c4a3019d-1ac0-4cfb-a838-2342eb992e6b",
    "tenant_id": "74610f3a5ad941998e91f076297ecf27",
    "device_id": "",
    "device_owner": "",
    "security_groups": [
      "93031677-2895-4b83-855a-637e309aa9e6"
    ],
  },
}
```

```
"extra_dhcp_opts": [],
"allowed_address_pairs": [],
"binding:vnic_type": "normal",
"binding:vif_details": {},
"binding:profile": {}
}
}
```

- Record the **port** ID in the response.

Step 2 Attach the NIC to the ECS.

- API

URI format: POST /v2.1/{tenant_id}/servers/{server_id}/os-interface

For details, see [5.7.3 Adding a NIC to an ECS](#).

- Example request

POST: <https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers/9f4d9281-95e7-4915-a126-1ee597101e2e/os-interface>

Obtain *{endpoint}* from [Regions and Endpoints](#).

Body:

```
{
  "interfaceAttachment": {
    "port_id": "7bf1c36f-e7f8-478a-be3d-674b486abbc4"
  }
}
```

- Example response

```
{
  "interfaceAttachment": {
    "port_state": "ACTIVE",
    "fixed_ips": [
      {
        "subnet_id": "877b5567-e8c6-4a0d-aabf-0f13da225fe5",
        "ip_address": "10.0.10.233"
      }
    ],
    "port_id": "7bf1c36f-e7f8-478a-be3d-674b486abbc4",
    "net_id": "c4a3019d-1ac0-4cfb-a838-2342eb992e6b",
    "mac_addr": "fa:16:3e:db:91:f6"
  }
}
```

Step 3 Verify the NIC attachment.

- API

URI format: GET /v2.1/{tenant_id}/servers/{server_id}/os-interface

For details, see [5.7.1 Querying ECS NICs](#).

- Example request

GET: <https://{endpoint}/v2.1/74610f3a5ad941998e91f076297ecf27/servers/9f4d9281-95e7-4915-a126-1ee597101e2e/os-interface>

Obtain *{endpoint}* from [Regions and Endpoints](#).

- Example response

```
{
  "interfaceAttachments": [
    {
      "port_state": "ACTIVE",
      "fixed_ips": [
        {
          "subnet_id": "46712fe4-25bd-4eae-874b-a528abfb76be",
          "ip_address": "192.168.0.50"
        }
      ]
    }
  ]
}
```



```
"version": "4",
"addr": "192.168.0.16",
"OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:37:de:ee",
"OS-EXT-IPS:type": "fixed",
"OS-EXT-IPS:port_id": "390b39b0-9a77-4ec2-ae1e-3af358f78999"
},
{
  "version": "4",
  "addr": "121.xx.xx.64",
  "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:37:de:ee",
  "OS-EXT-IPS:type": "floating",
  "OS-EXT-IPS:port_id": "390b39b0-9a77-4ec2-ae1e-3af358f78999"
}]
},
"flavor": {
  "disk": "0",
  "vcpus": "2",
  "ram": "4096",
  "id": "c6s.large.2",
  "name": "c6s.large.2"
},
"accessIPv4": "",
"accessIPv6": "",
"status": "SHUTOFF",
"progress": null,
"hostId": "604599c4e4eaa05d8865749e4c97979e14d74c6639a08460051b3a97",
"updated": "2021-02-18T12:38:39Z",
"created": "2021-02-18T12:37:42Z",
"metadata": {
  "metering.image_id": "6674d782-54ba-4f04-896d-95edd50f2eb9",
  "metering.imagetype": "gold",
  "metering.resourcespeccode": "c6s.large.2.linux",
  "image_name": "CentOS 8.2 64bit",
  "os_bit": "64",
  "cascaded.instance_extrainfo": "stopped_release_resource:True,pcibridge:1",
  "metering.resourcetype": "1",
  "vpc_id": "24bbb54c-659f-4141-8db9-a957e12b6ee8",
  "os_type": "Linux",
  "charging_mode": "0",
  "__support_agent_list": "ces"
},
"tags": [],
"description": "",
"locked": false,
"config_drive": "",
"tenant_id": "0b3ade290700f3612f29c005b9d16666",
"user_id": "0b3ade2a03800fec1f20c005d6116666",
"key_name": null,
"os-extended-volumes:volumes_attached": [{
  "device": "/dev/vda",
  "bootIndex": "0",
  "id": "0dc13ef4-dcf6-49d2-8d34-395d94767917",
  "delete_on_termination": "true"
}],
"OS-EXT-STS:task_state": null,
"OS-EXT-STS:power_state": 4,
"OS-EXT-STS:vm_state": "stopped",
"OS-EXT-SRV-ATTR:host": "604599c4e4eaa05d8865749e4c97979e14d74c6639a08460051b3a97",
"OS-EXT-SRV-ATTR:instance_name": "instance-003ef12a",
"OS-EXT-SRV-ATTR:hypervisor_hostname":
5eddb1b44af14ebaaa784cfba010f78f113b1fd0865fef854c264a925",
"OS-DCF:diskConfig": "MANUAL",
"OS-EXT-AZ:availability_zone": "cn-east-3c",
"os.scheduler_hints": {
},
"OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
"OS-EXT-SRV-ATTR:ramdisk_id": "",
"enterprise_project_id": "0",
```


7 Data Structure

7.1 Data Structure for Creating ECSs

Notes

ECS APIs can be of V1 or V1.1. V1 APIs can only be used to create pay-per-use ECSs, while V1.1 APIs can be used to create both pay-per-use and yearly/monthly ECSs.

For the fields described in this section, use V1.1 APIs for yearly/monthly ECSs.

Contents

- [publicip Field Description](#)
- [security_groups Field Description](#)
- [eip Field Description](#)
- [bandwidth Field Description](#)
- [extendparam Field Description for Assigning EIPs](#)
- [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](#)
- [server_tags Field Description](#)

publicip Field Description

This field is used by the following APIs:

- Creating ECSs `/v1/{project_id}/cloudservers`
- Creating ECSs: `/v1.1/{project_id}/cloudservers`

Table 7-1 publicip field description

Parameter	Mandatory	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.
eip	No	Object	Specifies an EIP that will be automatically assigned to an ECS. For details, see Table 7-3 .

 **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
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-2 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
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-3 eip field description

Parameter	Mandatory	Type	Description
iptype	Yes	String	Specifies the EIP type. For details, see the publicip field description in Assigning an EIP .
bandwidth	Yes	Object	Specifies the EIP bandwidth. For details, see bandwidth Field Description .

bandwidth Field Description

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-4 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 300. The minimum increment for bandwidth adjustment varies depending on the bandwidth range. <ul style="list-style-type: none">• The minimum increment is 1 Mbit/s if the allowed bandwidth ranges from 0 Mbit/s to 300 Mbit/s (with 300 Mbit/s included).• The minimum increment is 50 Mbit/s if the allowed bandwidth ranges from 300 Mbit/s to 1000 Mbit/s (with 1000 Mbit/s included).• The minimum increment is 500 Mbit/s if the allowed bandwidth is greater than 1000 Mbit/s. NOTE This parameter is mandatory when sharetype is set to PER and is optional when sharetype is set to WHOLE with an ID specified.

Parameter	Mandatory	Type	Description
sharetype	Yes	String	Specifies the bandwidth sharing type. Enumerated values: PER (indicates exclusive bandwidth) and WHOLE (indicates sharing)
chargemode	No	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 Assigning EIPs

This field is used by the following API:

Creating ECSs /v1.1/{project_id}/cloudservers

Table 7-5 extendparam field description for assigning EIPs

Parameter	Mandatory	Type	Description
chargingMode	No	String	Specifies the billing mode of an EIP. Options: <ul style="list-style-type: none"> prePaid: indicates the yearly/monthly billing mode. postPaid: indicates the pay-per-use billing mode. NOTE If sharetype in the bandwidth parameter with an ID specified is set to WHOLE , only pay-per-use EIPs are allowed and parameter prePaid is unavailable.

extendparam Field Description for Creating Disks

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-6 extendparam field description for creating disks

Parameter	Mandatory	Type	Description
resourceSpec Code	No	String	Specifies the code of the disk specifications, such as SATA, SAS, or SSD. NOTE This field has been discarded.
resourceType	No	String	Specifies the resource type. NOTE This field has been discarded.

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:</p> <p>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>When an ECS is created using a full-ECS image, the system automatically restores data from the data disk (if any) in the image. However, the disk type will be restored to the default settings: common I/O, VBD, and non-shared. 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 set snapshotId for each original data disk. Otherwise, the original data disks without snapshotId specified will use the default settings. 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. <p>Working rules:</p> <p>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:</p> <p>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> <p>If you have obtained the full-ECS image ID, obtain the Cloud Backup</p>

Parameter	Mandatory	Type	Description
			<p>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
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-7 extendparam field description for creating ECSs (for V1 APIs)

Parameter	Mandatory	Type	Description
chargingMode	No	Integer	<p>Specifies the billing mode.</p> <ul style="list-style-type: none"> • 0: indicates the pay-per-use billing mode. The default value is 0.
regionID	No	String	<p>Specifies the ID of the region where the ECS resides.</p> <p>See Regions and Endpoints.</p>

Parameter	Mandatory	Type	Description
support_auto_recovery	No	Boolean	<p>Specifies whether automatic recovery is enabled on the ECS.</p> <ul style="list-style-type: none"> • true: enables this function. • false: disables this function. <p>NOTE This parameter is of boolean type. If a non-boolean character is imported, the parameter value is set to false.</p> <p>When support_auto_recovery is set to false and "cond:compute": autorecovery is unavailable in the flavor, automatic recovery is not supported.</p> <p>When support_auto_recovery is set to false and "cond:compute": autorecovery is available in the flavor, automatic recovery is supported.</p> <p>You can query whether "cond:compute": autorecovery is available in the flavor by referring to 4.4.1 Querying Details About Flavors and Extended Flavor Information.</p>
CB_CSBS_BACKUP	No	String	<p>Specifies a CSBS policy ID and CSBS vault ID.</p> <p>For example, a CSBS policy ID obtained on the console is fdcaa27d-5be4-4f61-afe3-09ff79162c04.</p> <p>A CSBS vault ID is 332a9408-463f-436a-9e92-78dad95d1ac4.</p> <p>The CB_CSBS_BACKUP value is <code>{"policy_id": "fdcaa27d-5be4-4f61-afe3-09ff79162c04", "vault_id": "332a9408-463f-436a-9e92-78dad95d1ac4"}</code>.</p>

Table 7-8 extendparam field description for creating ECSs (for V1.1 APIs)

Parameter	Mandatory	Type	Description
chargingMode	No	String	Specifies the billing mode. Options: <ul style="list-style-type: none">• prePaid: indicates the yearly/monthly billing mode.• postPaid: indicates the pay-per-use billing mode.• The default value is postPaid. NOTE When chargingMode is set to prePaid (indicating that the created ECS is billed in yearly/monthly payments) and the ECS is logged in using an SSH key, op_svc_userid in metadata is mandatory.
regionID	No	String	Specifies the ID of the region where the ECS resides. See Regions and Endpoints .
periodType	No	String	Specifies the subscription period. Options: <ul style="list-style-type: none">• month: indicates that the subscription is in the unit of month.• year: indicates that the subscription is in the unit of year. NOTE This parameter is valid and mandatory if chargingMode is set to prePaid .
periodNum	No	Integer	Specifies the number of subscription periods. Options: <ul style="list-style-type: none">• If periodType is month, the value ranges from 1 to 9.• If periodType is year, the value ranges from 1 to 3. NOTE <ul style="list-style-type: none">• This parameter is valid and mandatory if chargingMode is set to prePaid.• The parameter value must be a positive integer.

Parameter	Mandatory	Type	Description
isAutoRenew	No	String	Specifies whether auto renew is enabled. <ul style="list-style-type: none">• true: indicates that auto renew is enabled.• false: indicates that auto renew is disabled. NOTE This parameter is valid when chargingMode is set to prePaid . If this parameter is not specified, auto renew is disabled by default.
isAutoPay	No	String	Specifies whether the order is automatically or manually paid. <ul style="list-style-type: none">• true: The order will be automatically paid.• false: The order must be manually paid. NOTE This parameter is valid when chargingMode is set to prePaid . If this parameter is not specified, the order must be manually paid by default.
support_auto_recovery	No	Boolean	Specifies whether to enable automatic ECS recovery. <ul style="list-style-type: none">• true: enables this function.• false: disables this function. NOTE This parameter is of boolean type. If a non-boolean character is imported, the parameter value is set to false .
marketType	No	String	Specifies a spot ECS. When creating a spot ECS, set the parameter value to spot . NOTE This parameter takes effect only when chargingMode is set to postPaid and marketType is set to spot .
spotPrice	No	String	Specifies the highest price per hour you accept for a spot ECS. NOTE <ul style="list-style-type: none">• This parameter takes effect only when chargingMode is set to postPaid and marketType is set to spot.• When chargingMode is set to postPaid and marketType is set to spot, if the spotPrice parameter is not specified or specified to null, the pay-per-use price is used by default.• The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price.

metadata Field Description for Creating Disks

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-9 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 .

metadata Field Description for Creating ECSs

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-10 metadata reserved field description

Parameter	Mandatory	Type	Description
op_svc_userid	No	String	Specifies the user ID. NOTE When chargingMode in the extendparam parameter is set to prePaid (indicating that the created ECS is billed in yearly/monthly payments) and the ECS is logged in using an SSH key, this field is mandatory.
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.

os:scheduler_hints Field Description

This field is used by the following APIs:

- Creating ECSs /v1/{project_id}/cloudservers
- Creating ECSs: /v1.1/{project_id}/cloudservers
- Creating ECSs (native API): /v2.1/{project_id}/servers

Table 7-11 os:scheduler_hints field description (request parameters)

Parameter	Mandatory	Type	Description
group	No	String	Specifies an ECS group ID, which is in UUID format. Obtain the parameter value from the console or by performing operations provided in 5.12.2 Querying ECS Groups . NOTE Ensure that the ECS group uses the anti-affinity policy. You are not advised to use other policies.
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 7-12 os:scheduler_hints field description (response parameters)

Parameter	Type	Description
group	Array of strings	Specifies an ECS group ID, which is in UUID format. Obtain the parameter value from the console or by performing operations provided in 5.12.2 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 .

server_tags Field Description

This field is used by the following APIs:

- Creating ECSs: /v1/{project_id}/cloudservers
- Creating ECSs: /v1.1/{project_id}/cloudservers

Table 7-13 server_tags field description

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

7.2 Data Structure for Querying Details About ECSs

Table 7-14 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 7-15 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.
ram	String	Specifies the memory size (MB) in the ECS flavor.

Table 7-16 security_groups parameters

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

The following table lists parameters involved in the fault information attribute.

Table 7-17 fault parameters

Parameter	Type	Description
message	String	Specifies the fault information.
code	Integer	Specifies the error code.
details	String	Specifies the fault details.
created	String	Specifies the time when the fault occurred. The time is in ISO 8601 time format.

Table 7-18 os-extended-volumes:volumes_attached parameters

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.

Parameter	Type	Description
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 newly added in microversion 2.3.
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 7-19 metadata parameters

Parameter	Type	Description
charging_mode	String	Specifies the ECS billing mode. <ul style="list-style-type: none">● 0: pay-per-use payment (postpaid)● 1: yearly/monthly payment (prepaid)● 2: spot price billing mode
metering.order_id	String	Specifies the order ID for a yearly/monthly ECS.
metering.product_id	String	Specifies the product ID for a yearly/monthly ECS.
vpc_id	String	Specifies the ID of the VPC where the ECS is located.
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: The value is gold.● Private image: The value is private.● Shared image: The value is shared.

Parameter	Type	Description
metering.resources peccode	String	Specifies the resource specifications of the ECS.
metering.resourcetype	String	Specifies the resource type of the ECS. Value 1 indicates ECSs.
cascaded.instance_ extrainfo	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.
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. <ul style="list-style-type: none">• Pay-per-use is changed to yearly/monthly (TO_PERIOD_LOCK)
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 7-20 sys_tags parameters

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

Table 7-21 image parameters

Parameter	Type	Description
id	String	Specifies the image ID.

7.3 Data Structure for Query 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 7-22 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• cpu1: computing I• cpu2: computing II• computingv3: general computing-plus• highmem: memory-optimized• saphana: large-memory HANA ECS• diskintensive: disk-intensive
resource_type	String	Specifies the resource type. resource_type is used to differentiate between the types of the physical servers accommodating ECSs.

Parameter	Type	Description
quota:local_disk	String	<p>The value of this parameter is in format of "{type}:{count}:{size}:{safeFormat}", where,</p> <ul style="list-style-type: none"> • type specifies the disk type, which can only be HDD. • count specifies the number of local disks. Its value can be 3/6/12/24 for d1 type of disks, 2/4/8/12/16/24 for d2 type of disks, or 2/4/8/12/16/24/28 for d3 type of disks. • size specifies 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 specifies whether the local disks of an ECS have been securely formatted. The value of this parameter can only be FALSE for D1 ECSs or True for D2 and D3 ECSs. <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 specification of the NVME SSD disk, which can be large or small. If the value is large, only I3 ECSs are supported. • num: indicates the number of partitions on the disk. • 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 spec value is small, the value of this parameter is 1/4 or 1/2 of the specification. • safeFormat: indicates whether the local disks of the ECS are securely formatted. If the value is True, only I3 ECSs are supported. <p>NOTE This field is dedicated for ultra-high I/O ECSs.</p>

Parameter	Type	Description
ecs:generation	String	<p>Specifies the generation of an ECS type.</p> <ul style="list-style-type: none"> • s1: general-purpose first-generation • s2: general-purpose second-generation • s3: general-purpose third-generation • s6: general-purpose • sn3: general network-optimized • c3: general computing-plus • c6: general computing-plus • c3ne: general network enhancement • m1: memory-optimized first-generation • m2: memory-optimized second-generation • m3: memory-optimized third-generation • m6: memory-optimized • m3ne: memory network enhancement • h1: high-performance computing first-generation • h2: high-performance computing second-generation • hc2: high-performance computing • h3: high-performance computing • hi3: ultra-high performance computing • d1: disk-intensive first-generation ECSs • d2: disk-intensive second-generation • d3: disk-intensive • kc1: Kunpeng general computing-plus • km1: Kunpeng memory-optimized • g1: GPU-accelerated first-generation • g2: GPU-accelerated second-generation • e3: large-memory • i3: ultra-high I/O <p>NOTE This field is optional.</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>

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 canceled (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 the recommended flavor (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 1000.
quota:min_rate	String	<p>Specified 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 1000.
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 10000, it must be divided by 10000.

Parameter	Type	Description
cond:operation:charge	String	Specifies a billing type. <ul style="list-style-type: none">• Both billing types are supported if this parameter is not set.• Yearly/Monthly• Pay-per-use
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.

8 Permissions Policies and Supported Actions

8.1 Lifecycle Management

Permission	API	Action	Dependent Permission
Creating ECSs	POST /v1.1/{project_id}/cloudservers	<ul style="list-style-type: none">Assigning a New EIP ecs:cloudServers:createUsing an Existing EIP ecs:cloudServers:create	<ul style="list-style-type: none">Assigning a New EIP vpc:publicIps:createUsing an Existing EIP vpc:publicIps:update
Creating an ECS (Pay-per-Use)	POST /v1/{project_id}/cloudservers	<ul style="list-style-type: none">Assigning a New EIP ecs:cloudServers:createUsing an Existing EIP ecs:cloudServers:create	<ul style="list-style-type: none">Assigning a New EIP vpc:publicIps:createUsing an Existing EIP vpc:publicIps:update
Deleting ECSs	POST /v1/{project_id}/cloudservers/delete	ecs:cloudServers:delete	N/A
Displaying Details About ECSs	GET /v1/{project_id}/cloudservers/detail	ecs:cloudServers:list	N/A

Permission	API	Action	Dependent Permission
Displaying Details About an ECS	GET /v1/{project_id}/cloudservers/{server_id}	ecs:cloudServers:get	N/A
Modifying an ECS	PUT /v1/{project_id}/cloudservers/{server_id}	ecs:cloudServers:put	N/A
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 ECSs (Native OpenStack API)	GET /v2.1/{project_id}/servers	ecs:servers:list	N/A

Permission	API	Action	Dependent Permission
Querying Details About an 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	Dependent Permission
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	N/A

Permission	API	Action	Dependent Permission
Modifying an ECS (Native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}	ecs:servers:update	ecs:servers:get

8.2 ECS Status Management

Permission	API	Action	Dependent Permission
Changing an ECS OS	POST /v2/{project_id}/cloudservers/{server_id}/changeos POST /v1/{project_id}/cloudservers/{server_id}/changeos	ecs:cloudServers:changeOS	N/A
Reinstalling an ECS OS	POST /v2/{project_id}/cloudservers/{server_id}/reinstallos POST /v1/{project_id}/cloudservers/{server_id}/reinstallos	ecs:cloudServers:rebuild	N/A
Modifying the Specifications of an ECS	POST /v1.1/{project_id}/cloudservers/{server_id}/resize	ecs:cloudServers:resize	N/A
Modifying the Specifications of an ECS (Pay-per-Use)	POST /v1/{project_id}/cloudservers/{server_id}/resize	ecs:cloudServers:resize	N/A
Cold Migrating an ECS	POST /v1/{project_id}/cloudservers/{server_id}/migrate	ecs:cloudServers:migrate	N/A

Permission	API	Action	Dependent Permission
Stopping an ECS (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:stop	ecs:servers:get
Restarting an ECS (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:reboot	ecs:servers:get
Modifying the Specifications of an ECS (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:resize	ecs:servers: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
Locking an ECS (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:lock	ecs:servers:get
Unlocking an ECS (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:unlock	ecs:servers:get

8.3 Batch Operations

Permission	API	Action	Dependent Permission
Stopping ECSs in a Batch	POST /v1/{project_id}/cloudservers/action	ecs:cloudServers:stop	N/A
Restarting ECSs in a Batch	POST /v1/{project_id}/cloudservers/action	ecs:cloudServers:reboot	N/A

Permission	API	Action	Dependent Permission
Starting ECSs in a Batch	POST /v1/{project_id}/cloudservers/action	ecs:cloudServers:start	N/A
Modifying ECSs in a Batch	PUT /v1/{project_id}/cloudservers/server-name	ecs:cloudServers:put	N/A
Attaching a Specified Shared EVS Disk to Multiple ECSs in a Batch	POST /v1/{project_id}/batchaction/attachvolumes/{volume_id}	ecs:cloudServers:attachSharedVolume	N/A

8.4 Network Management

Permission	API	Action	Dependent Permission
Querying Networks (Native OpenStack API)	GET /v2.1/{project_id}/os-networks	ecs:networks:list	vpc:networks:get

8.5 Image Management

Permission	API	Action	Dependent Permission
Creating an Image (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:servers:createImage	evs:volumes:get evs:snapshots:create ims:images:create ims:images:get ims:images:list ims:images:update ims:images:delete

8.6 Security Group Management

Permission	API	Action	Dependent Permission
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	Dependent Permission
Updating Information About 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 to Which an ECS Belongs (Native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-security-groups	ecs:securityGroups:use	vpc:securityGroups:get
Adding a Security Group (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:securityGroups:use	ecs:servers:get vpc:securityGroups:get vpc:securityGroups:create vpc:securityGroups:update vpc:ports:get vpc:ports:update
Removing a Security Group (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/action	ecs:securityGroups:use	ecs:servers:get vpc:securityGroups:get vpc:securityGroups:delete vpc:securityGroups:update vpc:ports:get vpc:ports:update

8.7 Specifications Query

Permission	API	Action	Dependent Permission
Querying Specifications and Expansion Details About ECSs	GET /v1/{project_id}/cloudservers/flavors	ecs:cloudServerFlavors:get	N/A

Permission	API	Action	Dependent Permission
Querying the Target ECS Flavors to Which a Flavor Can Be Changed	GET /v1/{project_id}/cloudservers/resize_flavors	ecs:cloudServers:list	N/A
Querying Details About extra_specs of an ECS (Native OpenStack API)	GET /v2.1/{project_id}/flavors/{flavors_id}/os-extra_specs	ecs:flavors:get	N/A

8.8 NIC Management

Permission	API	Action	Dependent Permission
Configuring a Private IP Address for a NIC of an ECS	PUT /v1/{project_id}/cloudservers/nics/{nic_id}	ecs:cloudServerNics:update	N/A
Deleting NICs from an ECS in a Batch	POST /v1/{project_id}/cloudservers/{server_id}/nics/delete	ecs:cloudServerNics:delete	N/A
Adding NICs to an ECS in a Batch	POST /v1/{project_id}/cloudservers/{server_id}/nics	ecs:cloudServers:addNics	N/A
Displaying NIC Information About ECSs	GET /v1/{project_id}/cloudservers/{server_id}/os-interface	ecs:cloudServers:get	N/A

Permission	API	Action	Dependent Permission
Adding an ECS NIC (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/os-interface	ecs:serverInterfaces:use	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	Dependent Permission
Deleting an ECS NIC (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 ECS NICs (Native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-interface	ecs:serverInterfaces:get	vpc:ports:get
Querying NIC Information About an ECS (Native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/os-interface/{id}	ecs:serverInterfaces:get	vpc:ports:get

8.9 Disk Management

Permission	API	Action	Dependent Permission
Detaching a Disk from a Specified ECS	DELETE /v1/{project_id}/cloudservers/{server_id}/detachvolume/{attachment_id}	ecs:cloudServers:detachVolume	N/A

Permission	API	Action	Dependent Permission
Attaching a Disk to an ECS	POST /v1/{project_id}/cloudservers/{server_id}/attachvolume	ecs:cloudServers:attach	N/A
Querying Information About Disks Attached to an ECS	GET /v1/{project_id}/cloudservers/{server_id}/block_device	ecs:cloudServers:get	N/A
Querying Disk Attachments of an ECS	GET /v1/{project_id}/cloudservers/{server_id}/os-volume_attachments	ecs:cloudServers:get	N/A
Querying a Single Disk Attached to an ECS	GET /v1/cloudservers/{server_id}/block_device/{volume_id}	ecs:cloudServers:get	N/A
Attaching a Disk from an ECS (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/os-volume_attachments	ecs:serverVolumeAttachments:create	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

Permission	API	Action	Dependent Permission
Querying Information About 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
Querying Information About 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 a Disk (Native OpenStack API)	GET /v2.1/{project_id}/os-volumes/{volume_id}	ecs:serverVolumes:use	evs:volumes:get
Querying Disks (Native OpenStack API)	GET /v2.1/{project_id}/os-volumes	ecs:serverVolumes:use	evs:volumes:get evs:volumes:list
Querying Detailed Disk Information (Native OpenStack API)	GET /v2.1/{project_id}/os-volumes/detail	ecs:serverVolumes:use	evs:volumes:get evs:volumes:list

8.10 Metadata Management

Permission	API	Action	Dependent Permission
Querying ECS Metadata (Native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/metadata	ecs:servers:listMetadata	N/A

Permission	API	Action	Dependent Permission
Querying Metadata of an ECS Key (Native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/metadata/{key}	ecs:servers:getMetadata	N/A
Deleting Specified ECS Metadata (Native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/metadata/{key}	ecs:servers:setMetadata	N/A
Modifying the Key Value in Metadata of an ECS (Native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}/metadata/{key}	ecs:servers:setMetadata	N/A
Updating ECS Metadata (Native OpenStack API)	POST /v2.1/{project_id}/servers/{server_id}/metadata	ecs:servers:setMetadata	N/A
Configuring ECS Metadata (Native OpenStack API)	PUT /v2.1/{project_id}/servers/{server_id}/metadata	ecs:servers:setMetadata	ecs:servers:get
Managing Automatic Recovery of an ECS	PUT /v1/{project_id}/cloudservers/{server_id}/autorecovery	ecs:cloudServers:setAutoRecovery	N/A
Querying Automatic Recovery of an ECS	GET /v1/{project_id}/cloudservers/{server_id}/autorecovery	ecs:cloudServers:getAutoRecovery	N/A

8.11 Tenant Quota Management

Permission	API	Action	Dependent Permission
Querying the Tenant Quota	GET /v1/{project_id}/cloudservers/limits	ecs:cloudServerQuotas:get	-
Querying Quotas of a Tenant (Native OpenStack API)	GET /v2.1/{project_id}/os-quotasets/{project_id}?user_id={user_id}	ecs:quotas:get	-

Permission	API	Action	Dependent Permission
Querying Default Quotas (Native OpenStack API)	GET /v2.1/{project_id}/os-quota-sets/{project_id}/defaults	ecs:quotas:get	-

8.12 SSH Key Management

Permission	API	Action	Dependent Permission
Creating and Importing an SSH Key Pair (Native OpenStack API)	POST /v2.1/{project_id}/os-keypairs	ecs:serverKeypairs:create	N/A
Querying an SSH Key Pair (Native OpenStack API)	GET /v2.1/{project_id}/os-keypairs/{keypair_name}	ecs:serverKeypairs:get	N/A
Querying SSH Key Pairs (Native OpenStack API)	GET /v2.1/{project_id}/os-keypairs	ecs:serverKeypairs:list	N/A
Deleting an SSH Key Pair (Native OpenStack API)	DELETE /v2.1/{project_id}/os-keypairs/{keypair_name}	ecs:serverKeypairs:delete	N/A

8.13 Password Management

Permission	API	Action	Dependent Permission
Resetting the Password for Logging In to an ECS with a Few Clicks for Enterprise Projects	PUT /v1/{project_id}/cloudservers/{server_id}/os-reset-password	ecs:cloudServers:resetServerPwd	N/A
Displaying Whether an ECS Supports Password Reset	GET /v1/{project_id}/cloudservers/{server_id}/os-resetpwd-flag	ecs:cloudServers:get	N/A

Permission	API	Action	Dependent Permission
Obtaining the Password for Logging In to a Windows ECS	GET /v1/{project_id}/cloudservers/{server_id}/os-server-password	ecs:cloudServers:get	N/A
Deleting the Password for Logging In to a Windows ECS	DELETE /v1/{project_id}/cloudservers/{server_id}/os-server-password	ecs:cloudServers:deletePassword	N/A
Resetting the Password for Logging In to an ECS with a Few Clicks	PUT /v2.1/{project_id}/servers/{server_id}/os-reset-password	ecs:cloudServers:resetServerPwd	N/A
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	N/A
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	N/A

8.14 Floating IP Address Management

Permission	API	Action	Dependent Permission
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

Permission	API	Action	Dependent Permission
Querying Details About Floating IP Addresses (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

8.15 ECS Group Management

Permission	API	Action	Dependent Permission
Deleting an ECS Group	DELETE /v1/{project_id}/cloudservers/os-server-groups/{server_group_id}	ecs:cloudServers:delete	N/A
Creating an ECS Group	POST /v1{project_id}/cloudservers/os-server-groups	ecs:cloudServers:create	N/A
Creating an ECS Group (Native OpenStack API)	POST /v2.1/{project_id}/os-server-groups	ecs:serverGroups:manage	N/A
Querying ECS Groups (Native OpenStack API)	GET /v2.1/{project_id}/os-server-groups	ecs:serverGroups:manage	N/A
Querying Details About an ECS Group (Native OpenStack API)	GET /v2.1/{project_id}/os-server-groups/{server_group_id}	ecs:serverGroups:manage	N/A
Deleting an ECS Group (Native OpenStack API)	DELETE /v2.1/{project_id}/os-server-groups/{server_group_id}	ecs:serverGroups:manage	N/A

8.16 ECS Management Through Console

Permission	API	Action	Dependent Permission
Obtaining the Address for Logging In to the Console Using VNC	POST /v2.1/{project_id}/servers/{server_id}/remote-consoles	ecs:servers:createConsole	ecs:servers:get
Obtaining the Address for Remotely Logging In to an ECS Using VNC	POST /v1/{project_id}/cloudservers/{server_id}/remote_console	ecs:cloudServers:vnc	-

8.17 AZ Management

Permission	API	Action	Dependent Permission
Querying AZs	GET /v1/{project_id}/availability-zones	ecs:cloudServers:list	-
Querying AZs (Native OpenStack API)	GET /v2.1/{project_id}/os-availability-zone	ecs:availabilityZones:list	-

8.18 Tag Management

Permission	API	Action	Dependent Permission
Adding or Deleting Tags to or from an ECS in a Batch	POST /v1/{project_id}/cloudservers/{server_id}/tags/action	ecs:cloudServers:put	N/A
Querying Project Tags	GET /v1/{project_id}/cloudservers/tags	ecs:cloudServers:list	N/A
Querying Tags of an ECS	GET /v1/{project_id}/servers/{server_id}/tags	ecs:servers:getTags	N/A

Permission	API	Action	Dependent Permission
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
Creating an ECS Tag (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
Querying an ECS Tag (Native OpenStack API)	GET /v2.1/{project_id}/servers/{server_id}/tags/{tag}	ecs:servers:getTags	ecs:servers:get
Deleting All ECS Tags (Native OpenStack API)	DELETE /v2.1/{project_id}/servers/{server_id}/tags	ecs:servers:setTags	ecs:servers:get

9 Common Parameters

9.1 Returned Values for General Requests

- Normal

Returned Value	Description
200	Request succeeded.
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
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.
406 Not Acceptable	The response generated by the server cannot be accepted by the client.

Returned Value	Description
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.

9.2 Error Code Description

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. When error messages are provided for the ECS-depended services, contact customer service 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 *Elastic Cloud Server User Guide* for troubleshooting.

Error Codes

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in [API Gateway Error Codes](#).

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs.0000	Request error.	An existing EIP cannot be assigned to the ECSs created in batches.	Check the request body according to the returned error message.
400	Ecs.0001	The number of ECSs has reached the maximum allowed.	The number of ECSs is beyond the quota limit.	Apply for a higher quota of the corresponding resource according to the returned error message.
400	Ecs.0002	Failed to submit the task.	Failed to submit the task.	Contact customer service to locate the fault.
400	Ecs.0003	You do not have permission or your balance is insufficient.	The token role contains op_suspended . This operation is not allowed.	Check whether the account balance is insufficient and the account is frozen according to the returned error message.
400	Ecs.0004	Authentication failed.	Failed to assign permissions: %s	For details, see the returned error message or contact customer service.
400	Ecs.0005	Invalid parameters.	Invalid request body.	Check whether the request body is of the correct JSON structure according to the API reference.
400	Ecs.0006	No product ID in the Marketplace image.	Marketplace image [%s] must have a product code.	Check image parameter.
400	Ecs.0007	Invalid image attributes.	ECS does not support Ironic image [%s].	Adjust the specifications or image type.
400	Ecs.0008	Invalid flavor attributes.	performancetype in extended flavor field [%s] is null.	Contact customer service to check whether the flavor registration is valid.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs.0009	Flavor conflict.	Change another flavor for resizing.	Change the flavor when modifying ECS specifications.
400	Ecs.0010	The private IP address is already in use.	Private IP address %s is already in use.	Change the port.
400	Ecs.0011	Failed to meet password complexity requirements.	The password must contain 8 to 26 characters.	Check the password length and change the password.
400	Ecs.0012	The number of IP addresses in the subnet is insufficient.	Insufficient IP addresses.	Check whether the floating IP addresses of the subnet are used up.
400	Ecs.0013	Insufficient EIP quota.	Insufficient EIP quota.	Apply for a higher EIP quota because the EIP quota is insufficient.
400	Ecs.0014	Invalid VPC parameters.	VPC parameters are invalid.	Check whether the subnets belong to the same VPC.
400	Ecs.0015	The disk of this type is not applicable to the ECS.	Flavor resource_type %s does not match volume_type %s.	Check whether the disk type is supported by the flavor.
400	Ecs.0017	The ECS is not the target one that the system disk or data disk to be attached.	ECS ID [%s] in disk metadata is different from ECS ID [%s].	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.	Failed to check the Cinder quota: The number of volumes is beyond the quota limit.	Apply for a higher EVS disk quota.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0022	The number of ECSs in the ECS group exceeded the upper limit.	The number of ECSs is beyond the ECS group quota limit.	Apply for a higher ECS quota for an ECS group.
400	Ecs. 0023	Invalid token, or the project ID in the token is different from that in the URL.	The project ID in token does not match that in 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.	Failed to check the KMS role.	Authorize EVS to obtain KMS keys for encrypting EVS disks.
400	Ecs. 0027	Private flavor, which cannot be used.	Flavor %s is private.	Change another flavor.
400	Ecs. 0028	The blacklisted user configured in the flavor is not allowed to use the flavor.	The user is contained in %s and is not allowed to use the flavor.	Change another flavor.
400	Ecs. 0029	The flavor does not exist or has been abandoned.	Flavor [%s] does not exist.	Change another flavor.
400	Ecs. 0030	The ECS has been frozen.	The ECS has been frozen.	Check whether the account has been frozen or contact customer service.
400	Ecs. 0031	The image does not exist.	Image [%s] does not exist.	Change another image.
400	Ecs. 0032	The image is not in Active state.	Image [%s] must be active.	Change another image.
400	Ecs. 0034	The full-ECS backup does not exist or has been deleted.	Backup %s does not exist.	Change another image.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs.0036	The flavor does not support automatic recovery.	Flavor [%s] does not support auto recovery.	Change another flavor.
400	Ecs.0037	The flavor does not support SCSI disks.	ECS flavor %s does not support SCSI disks.	Change another flavor or type.
400	Ecs.0038	The subnet does not exist.	Subnet [%s] does not exist.	Adjust network parameter settings.
400	Ecs.0039	The specified IP address does not belong to the subnet.	Private IP address [%s] is not in subnet [%s].	Change the specified private IP address.
400	Ecs.0041	Invalid description field.	Description field length cannot be greater than 85 characters and cannot contain characters ><.	Modify the service description field.
400	Ecs.0042	The number of attached data disks exceeds the maximum allowed limit.	The number of VBD disks is %s, but a KVM ECS supports a maximum of 24.	Adjust the number of attached data disks.
400	Ecs.0043	The disk type does not exist.	Disk type [%s] does not exist.	Change the disk type.
400	Ecs.0044	The disk of this type has been sold out.	Disk type [%s] has been sold out in AZ [%s].	Change the disk type.
400	Ecs.0045	The bandwidth exceeds the maximum size allowed.	Bandwidth size %d is invalid.	Adjust the bandwidth.
400	Ecs.0046	The disk type of the ECS is different from that of the snapshot image.	The root disk type in the request is different from that of the snapshot image.	Change the disk type.
400	Ecs.0048	The full-ECS image is unavailable.	Image [%s] is error or associated backup [%s] is error.	Check the full-ECS image.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs.0050	The number of NICs attached to the ECS exceeds the maximum value allowed.	The requested number of NICs is greater than the number available.	Adjust the number of NICs.
400	Ecs.0051	The attached disk is not of SCSI type.	Only SCSI disks can be attached to the ECSs.	Adjust the disk type.
400	Ecs.0052	The attached system disk is not of SCSI type.	Only SCSI system disks can be attached to the ECSs.	Change the system disk type.
400	Ecs.0053	The attached data disk is not of SCSI type.	Only SCSI data disks can be attached to the ECSs.	Change the data disk type.
400	Ecs.0057	The disk has been attached to the ECS.	The disk has been attached to the ECS and cannot be attached again.	Attach a new disk to the ECS.
400	Ecs.0058	The providerId of the image does not match the account ID.	The image provider does not match the account ID.	Check the account permission and image.
400	Ecs.0062	The flavor does not allow to set the NIC driver type.	The flavor does not support the driver mode.	Change another flavor.
400	Ecs.0064	Inconsistent VPC ID in the request body from that in the primary NIC.	The VPC ID in the request body is different from that in the primary NIC.	Adjust the NIC parameter settings.
400	Ecs.0073	The system disk is being backed up.	The disk cannot be deleted because it is being backed up.	You are not allowed to delete a system disk that is being backed up.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0074	External users are not allowed to create Windows ECSs.	Windows images are unavailable to external users.	External users, including non-internal users and non-third-party users, are not allowed to purchase Windows images.
400	Ecs. 0075	Partners can purchase only Windows images.	Only Windows images are available to partners.	Purchase only Windows images.
400	Ecs. 0100	The ECS status does not meet requirements.	Disks can be attached to ECS [%s] only in started or stopped state.	The ECS in the current state does not support this operation. Try again later.
400	Ecs. 0101	Abnormal system disk status.	Status error of the system disk.	For details, contact customer service.
400	Ecs. 0102	The system disk status does not allow the disk to be detached.	The status of the system disk attached to ECS [%s] does not support uninstallation.	Check the system disk status.
400	Ecs. 0103	The disk is unavailable.	Disk %s has been frozen and cannot be attached.	Check the disk status or contact customer service to change the disk status.
400	Ecs. 0104	Insufficient ECS disk quota for attaching more disks.	The number of disks attached to ECS [%s] has exceeded the upper limit.	Adjust the number of attached disks.
400	Ecs. 0105	Failed to query the ECS system disk.	Failed to view the details about the system disk: %s.	Check whether the ECS has a system disk attached.
400	Ecs. 0106	Abnormal network status.	Failed to create VLAN network %s because the network status is error.	For details, contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
403	Ecs. 0110	Operations are prohibited on the client due to permissions.	Token role %s is not allowed to perform this operation.	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 disk is not in the attachment list.	Disk %s is not in the attachment list for ECS %s.	Check whether the selected disk has been attached to the ECS, or replace the disk.
400	Ecs. 0112	The ECS is not of pay-per-use type, and it cannot be migrated.	Create a pay-per-use ECS.	For details, contact customer service.
404	Ecs. 0114	The ECS cannot be detected.	ECS [%s] could not be found.	Check whether the ECS has been created.
400	Ecs. 0118	The number of tasks in a batch is greater than the upper limit.	The number of ECSs %s is greater than the maximum number %s.	Check the number of ECSs in the batch.
400	Ecs. 0119	An ECS cannot be attached with an encrypted disk with a disabled key.	The key of the encrypted disk has been disabled, and the disk cannot be attached to the ECS.	Change the key status.
400	Ecs. 0120	The yearly/monthly ECS is not allowed to rebuild.	The yearly/monthly ECS cannot be rebuilt.	Do not rebuild yearly/monthly ECSs. Contact customer service for more details.
400	Ecs. 0121	Failed to attach the disk because the ECS and the disk are in different failure domains.	Disk must be in the same failure domain as that of the ECS.	Select a disk that is in the same failure domain as that of the target ECS.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0201	Failed to create the NIC.	Failed to create port in network %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0202	Failed to create the system disk.	Failed to create volume %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0203	Failed to create the data disk.	Failed to create disk %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0204	Failed to create the ECS.	Failed to add a tag to ECS %s: %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0205	Failed to attach the data disk.	Failed to call the Nova API to attach disk %s to ECS %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0207	Failed to modify ECS specifications.	Failed to resize ECS %s: %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0208	Failed to update the image metadata.	Failed to update the metadata of image %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0209	Failed to confirm the ECS specifications modification.	Failed to confirm the flavor change of ECS %s: %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0210	Failed to assign the floating IP address.	Failed to call the VPC API to assign an FIP to port %s: %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0211	Failed to create the NIC.	Failed to create QoS because %s.	For details, see the returned error message or contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0212	Failed to allocate the private IP address.	Failed to call the Neutron API to view private IP addresses because the response is null or invalid.	For details, contact customer service.
400	Ecs. 0213	Failed to update the port attributes.	Failed to update allowed_address_pairs of port %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0214	Failed to create the network.	Failed to create VLAN network because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0216	Failed to create the subnet.	Failed to create the subnet for VLAN %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0217	Failed to attach the NIC.	Failed to attach to ECS [%s] port [%s] because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0219	Failed to create the ECS.	Failed to quickly create ECS %s because of the error state or %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0221	Failed to migrate the ECS.	Cannot cold migrate from DeH [%s] to the same DeH.	For details, see the returned error message or contact customer service.
400	Ecs. 0226	Failed to start the ECS.	Failed to perform [%s] on ECS [%s]: [%s, %s]	For details, see the returned error message or contact customer service.
400	Ecs. 0301	Failed to query the ECS.	The information, status, or metadata of ECS %s is null.	For details, see the returned error message or contact customer service.
400	Ecs. 0302	Failed to query the ECS quota of the tenant.	Failed to view the quota usage of tenant %s because %s.	For details, see the returned error message or contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0303	Failed to query the flavor.	Failed to view flavor %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0304	Failed to query the image.	Failed to view image %s because the image or image name is null.	Contact customer service to check whether the image has been correctly registered or to check other causes.
400	Ecs. 0306	Failed to query the backup.	Failed to view the backup because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0307	Failed to query the port.	Failed to view the port because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0308	Failed to query the ECS quota of the tenant.	Failed to view limits because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0309	Failed to query the NIC QoS.	Failed to view QoS because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0310	Failed to view the network information.	Failed to view the network because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0311	Failed to query the disk type.	Failed to view the disk type of tenant %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0313	Failed to query the ECS group.	Failed to query ECS group: %s	For details, see the returned error message or contact customer service.
400	Ecs. 0314	Failed to obtain the key pair.	Failed to obtain the key pair because %s.	For details, see the returned error message or contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0315	Failed to obtain the automatic recovery status.	Failed to call the Nova API to obtain the automatic recovery status of tenant ID [%s] on ECS [%s]. The response is null or invalid.	For details, see the returned error message or contact customer service.
400	Ecs. 0319	Insufficient flavor capacity.	Check capacity: Capacity is not enough.	Apply for expanding the flavor capacity.
400	Ecs. 0320	Failed to obtain AZs.	Failed to obtain AZs: The response is null.	For details, see the returned error message or contact customer service.
400	Ecs. 0401	Failed to undo the operation performed on the port.	Failed to delete the port because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0402	Failed to undo the operation performed on the system disk.	Failed to delete system disk [%s] because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0403	Failed to undo the operation performed on the ECS.	Failed to delete ECS [%s] because %s.	Contact customer service to locate the fault.
400	Ecs. 0405	Failed to undo the operation performed on the data disk.	Failed to delete disk %s because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0501	Failed to delete the ECS.	Failed to delete ECS %s because downloading the system disk data is in progress.	Try again later.
400	Ecs. 0502	Failed to delete the private IP address.	Failed to roll back the EIP [%s] unbinding: %s.	For details, see the returned error message or contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0503	Failed to query the system disk.	Failed to view details about the disk because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 0507	Failed to delete the NIC.	Resource VLAN NICs cannot be deleted.	Check the NIC type.
400	Ecs. 0510	Yearly/Monthly ECSs do not support changing OSs.	The yearly/monthly ECS created using a Marketplace image does not support changing OS.	The ECSs created using a Marketplace image and billed on a yearly/monthly basis do not support changing OSs.
400	Ecs. 0513	Common users are not allowed to delete yearly/monthly ECSs.	ECS %s is in yearly/monthly payments and cannot be deleted by common users.	Unsubscribe from the ECS.
501	Ecs. 0603	Other commands are being executed. Try again 1 minute later.	The running state %s of ECS %s for tenant %s is unstable.	Try again 1 minute later.
400	Ecs. 0605	The ECS is locked.	ECS [%s] has been locked.	Check whether the ECS is locked. If so, unlock it.
400	Ecs. 0610	Resetting the password failed.	Failed to reset the password for logging in to ECS [%s].	Try again later or contact customer service.
400	Ecs. 0611	Requesting for a batch operation failed.	Batch operation failed.	Rectify the fault based on the returned error information and submit the request again.
400	Ecs. 0612	Failed to check whether plug-ins have been installed on an ECS.	Failed to check whether plug-ins have been installed on ECS [%s].	Try again later or contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0613	No plug-ins have been installed on the ECS.	ECS [%s] has no plug-ins installed.	Install desired plug-ins.
404	Ecs. 0614	The ECS cannot be detected.	itemNotFound: ECS xxx could not be found.	Check whether the ECS exists.
500	Ecs. 0615	An error has occurred in the request from an ECS.	The thread list is empty.	An internal system error occurred. Contact customer service to locate the fault.
400	Ecs. 0616	Failed to modify the ECS.	Failed to change the name of ECS [%s].	Try again later or contact customer service.
400	Ecs. 0617	Failed to modify the attributes of the disk attached to an ECS.	Failed to modify the attributes of the disk %s attached to ECS %s: %s	For details, see the returned error message or contact customer service.
400	Ecs. 0706	Failed to combine or split reserved instances.	Calling CBC to combine or split RI failed: %s	Contact customer service to locate the fault.
400	Ecs. 0707	The product does not exist.	the product [%s] does not exist	Contact customer service to locate the fault.
400	Ecs. 0811	The flavor cannot be switched from Xen to KVM.	The flavor cannot be resized to a new one.	Install a driver script.
400	Ecs. 0901	Yearly/Monthly DeHs are not supported.	Failed to create the ECS because yearly/Monthly ECSs cannot be created on DeHs.	Change another flavor.
400	Ecs. 0902	Spot ECSs do not support Marketplace images.	Marketplace images cannot be used to create spot ECSs.	Change another image.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0903	Spot ECSs do not support automatic recovery.	Spot ECSs do not support automatic recovery.	Change another flavor.
400	Ecs. 0904	UEFI images cannot be used to create Xen ECSs.	UEFI images cannot be used to create Xen ECSs.	Change another flavor.
400	Ecs. 0905	The number of tags exceeds the maximum number allowed.	The number of tags cannot be greater than 10.	Decrease the number of tags.
400	Ecs. 0906	Invalid tag attribute.	Invalid tag key.	Create a tag again.
400	Ecs. 0907	Invalid tag character set.	Tag key [%s] contains invalid characters.	Create a tag again.
400	Ecs. 0908	Duplicate tag key.	The tag key cannot be duplicate.	Create a tag again.
400	Ecs. 0909	The flavor does not support the disk type.	Flavor %s does not support disk type %s.	Change the flavor or disk type.
400	Ecs. 0910	Invalid NIC parameters for creating a HANA ECS.	The NIC parameters for creating a HANA ECS are invalid.	Adjust the NIC parameter settings.
400	Ecs. 0911	Invalid dedicated storage type of the disk.	Disks cluster type is different.	Modify parameter settings for the dedicated storage type.
400	Ecs. 0912	Invalid disk encryption attribute.	Encrypted key ID [%s] contains invalid characters.	Modify parameter settings for the disk encryption attribute.
400	Ecs. 0913	The number of ECSs to be created exceeds the maximum allowed limit.	Invalid number of ECSs.	Decrease the number of ECSs to be created.

HTTP Status Code	Error Code	Description	Error Message	Solution
400	Ecs. 0914	The length of the ECS name exceeds the maximum allowed limit.	Invalid length of ECS [%d].	Change the ECS name.
400	Ecs. 0915	The ECS name contains invalid characters.	ECS name [%s] contains invalid characters.	Change the ECS name.
400	Ecs. 0919	The port does not allow attaching.	Port [%s] in [%s] state does not allow attaching.	Change the port.
400	Ecs. 1000	Failed to call the Nova API.	Failed to call the Nova API because %s.	Internal calling error. Try again later or contact customer service.
404	Ecs. 1000	The resource does not exist.	itemNotFound: xxx could not be found.	Check whether the resource exists.
400	Ecs. 1001	OpenStack access error.	Failed to delete the ECS because the ECS is being deleted.	For details, contact customer service.
400	Ecs. 1002	OpenStack access timed out.	System timed out.	For details, contact customer service.
400	Ecs. 1100	Failed to access IAM.	Failed to call the IAM API because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 1200	Failed to access the VPC.	Failed to view the EIP because %s.	For details, see the returned error message or contact customer service.
400	Ecs. 1201	VPC access timed out.	Operation timed out.	For details, contact customer service.
400	Ecs. 1300	EVS access timed out.	Failed to call the Cinder API to create a disk because %s.	For details, see the returned error message or contact customer service.
403	Pdp. 0001	API authentication failed.	Policy does not allow %s to be performed.	Add permissions on IAM. For details, see API permissions.

HTTP Status Code	Error Code	Description	Error Message	Solution
202	Common. 0024	Limited by traffic control.	The traffic has exceeded the upper limit.	The number of concurrent requests has exceeded the upper limit. Try again later.
400	Common. 0002	Empty request body.	Request body is null!	Check the request body.
400	Common. 0011	Invalid job ID.	Failed to obtain the job.	Check whether the source of the job ID is correct.
400	Common. 0018	Invalid token, or the project ID in the token is different from that in the URL.	Tenant ID in token is not the same as that in URL.	Check whether the tenant token is correct.
400	Common. 0020	Failed to retry the task.	Failed to call the redo API.	Contact customer service.
400	Common. 0021	An error has occurred in job query.	Failed to obtain the job.	Try again later or contact customer service.
400	Common. 0022	An error has occurred in job submission.	Job failed.	Contact customer service.
400	Common. 0999	Task terminated.	The system was down.	Contact customer service.
400	Common. 0025	An error has occurred in task query.	Failed to obtain the task because %s.	Try again later or contact customer service.
400	Common. 0026	An error occurred in AZ query.	Fail to obtain the AZ.	Try again later or contact customer service.
401	Common. 0013	Invalid token.	Do not perform this operation.	Check whether the tenant token is correct.
500	Common. 0001	A system exception occurred.	System error.	Contact customer service.

HTTP Status Code	Error Code	Description	Error Message	Solution
503	Common. 1503	Limited by API traffic control.	An error has occurred in API traffic control because %s.	Too many APIs are being executed. Try again later.

9.3 HTTP Status Code

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
500	Internal Server Error
501	Not Implemented
503	Service Unavailable

9.4 Obtaining a Project ID

Scenarios

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

- [Obtain the Project ID by Calling an API](#)
- [Obtain the Project ID from the Console](#)

Obtain the Project ID by Calling an API

You can obtain the project ID by calling the API used to [query project information based on specified criteria](#).

The API used to obtain a project ID is GET `https://{Endpoint}/v3/projects`. {Endpoint} is the IAM endpoint and can be obtained from [Regions and Endpoints](#).

For details about API authentication, see [3.2 Authentication](#).

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

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

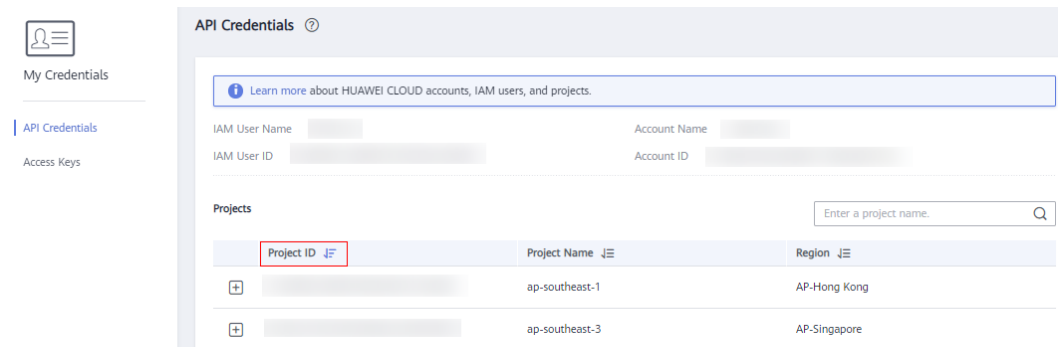
Obtain a Project ID from the Console

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

1. Log in to the management console.
2. Click the username and select **My Credentials** from the drop-down list.

On the **API Credentials** page, view the project ID in the project list.

Figure 9-1 Viewing the project ID



9.5 Task Request Result

9.5.1 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 4.9 Task Status Management .

- Abnormal response description

Parameter	Type	Description
error	Object	Specifies the returned error message when an error occurs. For details, see Table 9-1 .

Table 9-1 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 9-2 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": "70a599e0-31e7-49b7-b260-868f441e862b"
}
```

Abnormal response

```
{
  "error": {"message": "", "code": "XXX",""}
}
```

Abnormal response containing error details:

```
{
  "error": {
    "message": "xxx",
    "code": "xxx",
    "details": [
      {
        "code": "xxx",
        "message": "xxx"
      }
    ]
  }
}
```

9.5.2 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.

9.6 Batch Task Request

9.6.1 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 [9.5.1 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 9-3 .

Table 9-3 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 9-4 .
internalError	Array of objects	Specifies the error in each request among the requests submitted in a batch. For details, see Table 9-5 .

Table 9-4 error field structure

Parameter	Type	Description
message	String	Describes a batch operation error.
code	String	Specifies the code for a batch operation error.

Table 9-5 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"
    }
  ]
}
```

10 Out-of-Date APIs

10.1 Image Management (OpenStack Nova APIs)

10.1.1 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 10-1](#) describes the parameters in the URI.

Table 10-1 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

NOTE

Pagination query is supported. For details, see [1.7 Querying Data in Pages](#).

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 10-2](#) describes the query parameters.

Table 10-2 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 varies depending on the status in Glance. Table 10-3 shows the mapping of image statuses 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 10-3 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 10-4](#) describes the response parameters.

Table 10-4 Response parameters

Parameter	Mandatory	Type	Description
images	Yes	Array of objects	Specifies the image information.

Parameter	Mandatory	Type	Description
images_links	No	Array of objects	Specifies the information about the next page when you query images in pages.

Table 10-5 images information

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 10-6 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 10-7 links parameter description

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
```

Example Response

```
{
  "images": [
    {
      "id": "ee10f19c-503c-44af-af2f-73d5e42f7a17",
      "links": [
        {
          "href": "http://172.25.150.84:8774/v2/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
          "rel": "self"
        },
        {
          "href": "http://172.25.150.84:8774/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
          "rel": "bookmark"
        },
        {
          "href": "http://172.25.150.84:9292/d9ebe43510414ef590a4aa158605329e/images/ee10f19c-503c-44af-af2f-73d5e42f7a17",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "name": "image1"
    }
  ]
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

10.1.2 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 10-8](#) describes the parameters in the URI.

Table 10-8 Path parameters

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

NOTE

Pagination query is supported. For details, see [1.7 Querying Data in Pages](#).

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 10-9](#) describes the query parameters.

Table 10-9 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 10-10 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 .

Parameter	Mandatory	Type	Description
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 10-10 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 10-11](#) describes the response parameters.

Table 10-11 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.

Parameter	Type	Description
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 10-12 links parameter description

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/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://172.25.150.84:8774/v2/d9ebe43510414ef590a4aa158605329e/images/
ee10f19c-503c-44af-af2f-73d5e42f7a17",
    "rel": "self"
  },
  {
    "href": "http://172.25.150.84:8774/d9ebe43510414ef590a4aa158605329e/images/
ee10f19c-503c-44af-af2f-73d5e42f7a17",
    "rel": "bookmark"
  },
  {
    "href": "http://172.25.150.84:9292/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://172.25.150.84:8774/v2/d9ebe43510414ef590a4aa158605329e/servers/
b600b5b1-ed8c-4814-aefa-8b903c894c20",
      "rel": "self"
    },
    {
      "href": "http://172.25.150.84:8774/d9ebe43510414ef590a4aa158605329e/servers/b600b5b1-
ed8c-4814-aefa-8b903c894c20",
      "rel": "bookmark"
    }
  ]
},
"status": "ACTIVE",
"updated": "2014-02-10T17:05:07Z"
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

10.1.3 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 10-13](#) describes the parameters in the URI.

Table 10-13 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
image_id	Yes	Specifies the image ID.

Request

None

Response

[Table 10-14](#) describes the response parameters.

Table 10-14 Response 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.
metadata	Yes	Object	Specifies the key pair of the metadata.
OS-EXT-IMG-SIZE:size	Yes	Integer	Specifies the image size. The value must be greater than zero.

Parameter	Mandatory	Type	Description
minDisk	Yes	Integer	Specifies the minimum disk size in GB required by the image. The value must be greater than zero.
minRam	Yes	Integer	Specifies the minimum memory size in GB required by the image. The value must be greater than zero.
progress	Yes	Integer	Specifies the image upload progress. The value must be greater than zero.
status	Yes	String	Specifies the image status.
created	Yes	String	Specifies the image creation time. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .
updated	Yes	String	Specifies the image update time. The value is in ISO8601 format, such as 2013-06-09T06:42:18Z .

Table 10-15 links parameter description

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/  
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/v2/719e9483f42d4784a089862ac4c3e8d0/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 [9.1 Returned Values for General Requests](#).

10.1.4 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 10-16 describes the parameters in the URI.

Table 10-16 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
image_id	Yes	Specifies the image ID.

Request

None

Response

Table 10-17 describes the response parameters.

Table 10-17 Response parameter

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:pcsimsbeta:suse12.2-addx710-05-11",
    "virtual_env_type": "Ironic",
    "__platform": "Suse",
    "__support_o3s": "true"
  }
}
```

Returned Values

See [9.1 Returned Values for General Requests](#).

10.1.5 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 10-18](#) describes the parameters in the URI.

Table 10-18 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 [9.1 Returned Values for General Requests](#).

10.2 Security Group Management (OpenStack Nova APIs)

10.2.1 Querying Security Groups (Discarded)

Function

This API is used to query security groups.

This API has been discarded. Use the API described in [Querying Security Groups](#).

URI

GET /v2.1/{project_id}/os-security-groups

[Table 10-19](#) describes the parameters in the URI.

Table 10-19 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

NOTE

Pagination query is not supported.

Request

N/A

Response

[Table 10-20](#) describes the response parameters.

Table 10-20 Response parameters

Parameter	Type	Description
security_groups	Array of objects	Specifies security groups. For details, see Table 10-21 .

Table 10-21 security_group objects

Parameter	Type	Description
description	String	Specifies information about a security group. It is a string of 0 to 255 characters.
id	String	Specifies the security group ID in UUID format.
name	String	Specifies the security group name. It is a string of 0 to 255 characters.
rules	Array of objects	Specifies security group rules. For details, see Table 10-22 .
tenant_id	String	Specifies the tenant or project ID.

Table 10-22 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>. port_range_min indicates the ICMP type field, and port_range_max indicates the ICMP code field.</p>

Parameter	Type	Description
to_port	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 , this parameter indicates the ICMP code field with a length from 0 to 255 characters. 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.
ip_range	Object	Specifies the peer IP segment in CIDR format. For details, see Table 10-23 . Specify either ip_range or group .
group	Object	Specifies the name of the peer security group and the ID of the tenant in the peer security group. For details, see Table 10-24 . Specify either ip_range or group .
id	String	Specifies the security group rule ID in UUID format.

Table 10-23 ip_range objects

Parameter	Type	Description
cidr	String	Specifies the peer IP segment in CIDR format.

Table 10-24 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
```

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

10.2.2 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 [Creating a Security Group](#).

URI

POST /v2.1/{project_id}/os-security-groups

[Table 10-25](#) describes the parameters in the URI.

Table 10-25 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 10-26](#) describes the request parameters.

Table 10-26 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 10-27 .

Table 10-27 Objects of request parameter **security_group**

Parameter	Mandatory	Type	Description
name	No	String	Specifies the security group name. It is a string of 0 to 255 characters.
description	No	String	Specifies information about a security group. It is a string of 0 to 255 characters.

Response

[Table 10-28](#) describes the response parameters.

Table 10-28 Response parameters

Parameter	Type	Description
security_group	Object	Specifies the security group. For details, see Table 10-29 .

Table 10-29 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 [9.1 Returned Values for General Requests](#).

10.2.3 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 [Querying a Security Group](#).

URI

GET /v2.1/{project_id}/os-security-groups/{security_group_id}

[Table 10-30](#) describes the parameters in the URI.

Table 10-30 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
security_group_id	Yes	Specifies the security group ID, which is specified in the URI.

Request

None

Response

[Table 10-31](#) describes the response parameters.

Table 10-31 Response parameters

Parameter	Type	Description
security_group	Object	Specifies the security group. For details, see Table 10-32 .

Table 10-32 security_group objects

Parameter	Type	Description
description	String	Specifies information about a security group. It is a string of 0 to 255 characters.
id	String	Specifies the security group ID in UUID format.
name	String	Specifies the security group name. It is a string of 0 to 255 characters.

Parameter	Type	Description
rules	Array of objects	Specifies security group rules. For details, see Table 10-33 .
tenant_id	String	Specifies the tenant or project ID.

Table 10-33 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	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 indicates the ICMP type field with a length from 0 to 255 characters. 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.
to_port	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 , this parameter indicates the ICMP code field with a length from 0 to 255 characters. 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.
ip_range	Object	Specifies the peer IP segment in CIDR format. For details, see Table 10-34 . Specify either ip_range or group .
group	Object	Specifies the name of the peer security group and the ID of the tenant in the peer security group. For details, see Table 10-35 . Specify either ip_range or group .

Parameter	Type	Description
id	String	Specifies the security group rule ID.

Table 10-34 ip_range objects

Parameter	Type	Description
cidr	String	Specifies the peer IP segment in CIDR format.

Table 10-35 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 [9.1 Returned Values for General Requests](#).

10.2.4 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 [Updating a Security Group](#).

URI

```
PUT /v2.1/{project_id}/os-security-groups/{security_group_id}
```

[Table 10-36](#) describes the parameters in the URI.

Table 10-36 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
security_group_id	Yes	Specifies the security group ID, which is specified in the URI.

Request

[Table 10-37](#) describes the request parameters.

Table 10-37 Request parameters

Parameter	Mandatory	Type	Description
security_group	Yes	Object	Specifies the security group in the message body. For details, see Table 10-38 .

Table 10-38 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 10-39](#) describes the response parameters.

Table 10-39 Response parameters

Parameter	Mandatory	Type	Description
security_group	Yes	Object	Specifies the security group. For details, see Table 10-40 .

Table 10-40 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 10-41 .
tenant_id	Yes	String	Specifies the tenant or project ID. The value cannot exceed 255 characters.

Table 10-41 **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. 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.

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 . 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 10-42 . 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 10-43 . The value of ip_range or group must be empty.
id	Yes	String	Specifies the security group rule ID in UUID format.

Table 10-42 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 10-43 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 [9.1 Returned Values for General Requests](#).

10.2.5 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 [Deleting a Security Group](#).

URI

DELETE /v2.1/{project_id}/os-security-groups/{security_group_id}

[Table 10-44](#) describes the parameters in the URI.

Table 10-44 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 [9.1 Returned Values for General Requests](#).

10.2.6 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 [Creating a Security Group Rule](#).

URI

POST /v2.1/{project_id}/os-security-group-rules

[Table 10-45](#) describes the parameters in the URI.

Table 10-45 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 10-46](#) describes the request parameters.

Table 10-46 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 10-47 .

Table 10-47 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 .
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 10-48](#) describes the response parameters.

Table 10-48 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 10-49 .

Table 10-49 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 .
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 10-50 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 [9.1 Returned Values for General Requests](#).

10.2.7 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 [Deleting a Security Group Rule](#).

URI

DELETE /v2.1/{project_id}/os-security-group-rules/{security_group_rule_id}

[Table 10-51](#) describes the parameters in the URI.

Table 10-51 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Parameter	Mandatory	Description
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 [9.1 Returned Values for General Requests](#).

10.3 Disk Management (OpenStack Nova APIs)

10.3.1 Querying Brief Information About Disks (Discarded)

Function

This API is used to query brief information about disks.

This API has been discarded. Use the API described in [Querying EVS Disks \(OpenStack Cinder API v2\)](#).

URI

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

[Table 10-52](#) describes the parameters in the URI.

Table 10-52 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

N/A

Response

[Table 10-53](#) describes the response parameters.

Table 10-53 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 10-54 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.

Parameter	Type	Description
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
```

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

10.3.2 Querying Detailed Information About Disks (Discarded)

Function

This API is used to query detailed information about disks.

This API has been discarded. Use the API described in [Querying Details About All Disks \(OpenStack Cinder API v2\)](#).

URI

```
GET /v2.1/{project_id}/os-volumes/detail
```

[Table 10-55](#) describes the parameters in the URI.

Table 10-55 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

N/A

Response

[Table 10-56](#) describes the response parameters.

Table 10-56 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 10-57 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 [9.1 Returned Values for General Requests](#).

10.3.3 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 API described in [Querying Details About a Disk \(OpenStack Cinder API v2\)](#).

URI

```
GET /v2.1/{project_id}/os-volumes/{volume_id}
```

[Table 10-58](#) describes the parameters in the URI.

Table 10-58 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
volume_id	Yes	Specifies the disk ID.

Request

None

Response

Table 10-59 describes the response parameters.

Table 10-59 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 10-60 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 [9.1 Returned Values for General Requests](#).

10.3.4 Creating a Disk (Discarded)

Function

This API is used to create a disk.

This API has been discarded. Use the API described in [Creating EVS Disks \(OpenStack Cinder API v2\)](#).

URI

POST /v2.1/{project_id}/os-volumes

[Table 10-61](#) describes the parameters in the URI.

Table 10-61 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 10-62](#) describes the request parameters.

Table 10-62 Request parameters

Parameter	Mandatory	Type	Description
availability_zone	No	String	Specifies the AZ to which the volume to be created belongs. If the specified AZ does not exist, creating the volume failed, and the volume is in error state. The AZ to which the volume to be created belongs must be specified in the public cloud system.
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 10-63](#) describes the response parameters.

Table 10-63 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 [9.1 Returned Values for General Requests](#).

10.3.5 Deleting a Disk (Discarded)

Function

This API is used to delete a specified disk.

This API has been discarded. Use the API described in [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 10-64](#) describes the parameters in the URI.

Table 10-64 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 [9.1 Returned Values for General Requests](#).

10.4 Floating IP Address Management (OpenStack Nova APIs)

10.4.1 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. Use the VPC API [Updating a Floating IP Address](#).

URI

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

[Table 10-65](#) describes the parameters in the URI.

Table 10-65 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 10-66](#) describes the request parameters.

Table 10-66 Request parameter

Parameter	Mandatory	Type	Description
addFloatingIp	Yes	Object	Specifies the floating IP address to be bound to an ECS.

Table 10-67 addFloatingIp parameter information

Parameter	Mandatory	Type	Description
address	Yes	String	Specifies the floating IP address.
fixed_addresses	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 [9.1 Returned Values for General Requests](#).

10.4.2 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. Use the VPC API [Updating a Floating IP Address](#).

URI

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

[Table 10-68](#) describes the parameters in the URI.

Table 10-68 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
server_id	Yes	Specifies the ECS ID.

Request

[Table 10-69](#) describes the request parameters.

Table 10-69 Request parameter

Parameter	Mandatory	Type	Description
removeFloatingIp	Yes	Object	Unbinds a floating IP address from an ECS.

Table 10-70 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 [9.1 Returned Values for General Requests](#).

10.4.3 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 10-71](#) describes the parameters in the URI.

Table 10-71 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 10-72](#) describes the request parameters.

Table 10-72 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 10-73](#) describes the response parameters.

Table 10-73 Response parameters

Parameter	Mandatory	Type	Description
floating_ip	Yes	Object	Specifies the floating IP address. For details, see Table 10-74 .

Table 10-74 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 [9.1 Returned Values for General Requests](#).

10.4.4 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 10-75](#) describes the parameters in the URI.

Table 10-75 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 10-76](#) describes the response parameters.

Table 10-76 Response parameters

Parameter	Mandatory	Type	Description
floating_ips	Yes	Array of objects	Specifies the floating IP addresses.

Table 10-77 floating_ip objects

Parameter	Mandatory	Type	Description
floating_ip	Yes	Object	Specifies the floating IP address.

Table 10-78 floating_ip attributes

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.

Parameter	Mandatory	Type	Description
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 [9.1 Returned Values for General Requests](#).

10.4.5 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 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 10-79](#) describes the parameters in the URI.

Table 10-79 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
floating_ip_id	Yes	Specifies the ID of the floating IP address.

Request

None

Response

[Table 10-80](#) describes the response parameters.

Table 10-80 Response parameters

Parameter	Mandatory	Type	Description
floating_ip	Yes	Object	Specifies the floating IP address. For details, see Table 10-81 .

Table 10-81 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

```
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 [9.1 Returned Values for General Requests](#).

10.4.6 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 10-82](#) describes the parameters in the URI.

Table 10-82 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 [9.1 Returned Values for General Requests](#).

10.4.7 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 10-83](#) describes the parameters in the URI.

Table 10-83 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

None

Response

[Table 10-84](#) describes the response parameters.

Table 10-84 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 [9.1 Returned Values for General Requests](#).

10.5 Snapshot Management (OpenStack Nova APIs)

10.5.1 Creating a Snapshot (Discarded)

Function

This API is used to create a snapshot for a volume.

This API has been discarded. Use the API described in [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 10-85](#) describes the parameters in the URI.

Table 10-85 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .

Request

[Table 10-86](#) describes the request parameters.

Table 10-86 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_ . Therefore, 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 for the volume in use can be created.

Response

[Table 10-87](#) describes the response parameters.

Table 10-87 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 [9.1 Returned Values for General Requests](#).

10.5.2 Querying Snapshots (Discarded)

Function

This API is used to query information about a volume snapshot.

This API has been discarded. Use the API described in [Querying Details About an EVS Snapshot \(OpenStack Cinder API v2\)](#).

URI

GET /v2.1/{project_id}/os-snapshots/{snapshot_id}

[Table 10-88](#) describes the parameters in the URI.

Table 10-88 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 Obtaining a Project ID .
snapshot_id	Yes	Specifies the volume snapshot ID.

Response

Response parameters

[Table 10-89](#) describes the response parameters.

Table 10-89 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.
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 [9.1 Returned Values for General Requests](#).

10.5.3 Deleting a Snapshot (Discarded)

Function

This API is used to delete a volume snapshot.

This API has been discarded. Use the API described in [Deleting an EVS Snapshot \(OpenStack Cinder API v2\)](#).

URI

DELETE /v2.1/{project_id}/os-snapshots/{snapshot_id}

Table 10-90 describes the parameters in the URI.

Table 10-90 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the ID, see 9.4 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 [9.1 Returned Values for General Requests](#).

A Appendix

A.1 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 and has started to perform the modification.
ACTIVE	The ECS is running properly.
SHUTOFF	The ECS has been properly stopped.
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.

Status	Description
DELETED	The ECS has been deleted.

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.

Table A-3 OS-EXT-STS:task_state statuses

Status	Description
scheduling	The ECS is being created.
block_device_m apping	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.
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_d evice_mapping	The ECS is being rebuilt, and disks are being prepared for the ECS.

Status	Description
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.

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

vm_state	task_state	status
	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
	deleting null	ERROR
deleted	null	DELETED

A.2 Network APIs

For details about network APIs, see [Virtual Private Cloud API Reference](#).

B Change History

Released On	Description
2021-04-08	<p>This issue is the fourteenth official release.</p> <p>Modified the following content:</p> <p>Added parameters tenancy and dedicated_host_id to the os:scheduler_hints field in 7.1 Data Structure for Creating ECSs.</p>
2021-03-03	<p>This issue is the thirteenth official release.</p> <p>Added the following content:</p> <p>Added the CB_CSBS_BACKUP field about CSBS policy and CSBS vault in 7.1 Data Structure for Creating ECSs.</p> <p>Modified the following content:</p> <p>Modified the value range of sort_key in 5.2.5 Querying Details About ECSs.</p>
2020-01-21	<p>This issue is the twelfth official release.</p> <p>Modified the following content:</p> <ul style="list-style-type: none">Added response parameters in 5.1.1 Querying All API Versions and 5.1.2 Querying a Specified API Version.
2019-08-31	<p>This issue is the eleventh official release.</p> <p>Added the following content:</p> <ul style="list-style-type: none">4.12.3 Querying ECS Groups4.12.4 Querying Details About an ECS Group <p>Modified the following content:</p> <ul style="list-style-type: none">Moved discarded APIs to 10 Out-of-Date APIs.Modified the volumetype field description in 4.1.1 Creating ECSs and 4.1.2 Creating an ECS (Pay-per-Use).

Released On	Description
2019-08-23	<p>This issue is the tenth official release.</p> <p>Modified the following content:</p> <ul style="list-style-type: none">• Added 6.2 Example 2: Querying ECSs.• Deleted the V2.1 URI in 4.2.1 Reinstalling an ECS OS (Using an Image with Cloud-Init Installed).• Deleted the V2.1 URI in 4.2.2 Changing an ECS OS (Using an Image with Cloud-Init Installed).

Released On	Description
2019-07-30	<p>This issue is the ninth official release.</p> <p>Added the following content:</p> <ul style="list-style-type: none">• Added version selection in 1.6 Selecting an API Type or Version.• 4.1.6 Modifying an ECS• 4.2.7 Cold Migrating an ECS• 4.2.8 Obtaining the VNC Login Address• 4.4.2 Querying the Target ECS Flavors to Which a Flavor Can Be Changed• 4.5.5 Query NICs of an ECS• 4.6.1 Querying a Single Disk Attached to an ECS• 4.6.3 Querying Information About Disks Attached to an ECS• 4.6.2 Querying Disk Attachments of an ECS• 4.7.1 Updating ECS Metadata• 4.7.2 Deleting Specified ECS Metadata• 4.10.4 Querying Project Tags• 4.10.5 Querying Tags of an ECS• 4.10.2 Adding Tags to an ECS in a Batch• 4.10.3 Deleting Tags from an ECS in a Batch• 4.11.1 Querying Whether One-Click Password Reset Is Supported• 4.11.2 Resetting the Password for Logging In to an ECS with a Few Clicks• 4.11.3 Retrieving the Password for Logging In to a Windows ECS• 4.11.4 Deleting the Password for Logging In to a Windows ECS• 4.12.1 Creating an ECS Group• 4.12.2 Deleting an ECS Group• 5.14.2 Obtaining a VNC-based Remote Login Address (Microversion 2.6 or Later)• Added V2.1 URIs in 4.2.1 Reinstalling an ECS OS (Using an Image with Cloud-Init Installed).• Added V2.1 URIs in 4.2.2 Changing an ECS OS (Using an Image with Cloud-Init Installed).• 6.3 Example 3: Modifying ECS Specifications• 6.4 Example 4: Attaching a Disk to an ECS• 6.5 Example 5: Attaching a NIC to an ECS <p>Modified the following content:</p>

Released On	Description
	<ul style="list-style-type: none"> • Modified the response messages and examples and added serverIds in 4.1.2 Creating an ECS (Pay-per-Use) and 4.1.1 Creating ECSs. • Modified the subnet_id field description in 4.1.2 Creating an ECS (Pay-per-Use) and 4.1.1 Creating ECSs. • Added error codes Ecs.0802 through Ecs.08010 in 9.2 Error Code Description. • Added error codes Ecs.0046 and Ecs.0048 through Ecs.0053 in 9.2 Error Code Description. • Added the metadata field for creating disks in 7.1 Data Structure for Creating ECSs. • Modified the response example in 4.2.10 Modifying the Specifications of an ECS (Pay-per-Use). • Added the enterprise_project_id field to V1.1 APIs in 7.1 Data Structure for Creating ECSs. • Deleted the following parameters in 5.2.5 Querying Details About ECSs because they are not returned by the API: evsOpts, hyperThreadAffinity, numaOpts, and vcpuAffinity. • Deleted the following parameters in 5.2.6 Querying Details About ECSs because they are not returned by the API: evsOpts, hyperThreadAffinity, numaOpts, and vcpuAffinity. • Added fault in 5.2.5 Querying Details About ECSs. • Added fault in 5.2.6 Querying Details About ECSs. • Canceled the limit_by_flavor field in 4.8.1 Querying Tenant Quotas. • Adjusted the table structure of API permissions.

Released On	Description
2019-05-30	<p>This issue is the eighth official release.</p> <p>Added the following content:</p> <ul style="list-style-type: none"> ● Deleted the V2 API URI and added the recommended V2.1 API in 5 OpenStack Nova APIs. ● 4.1.1 Creating ECSs ● 4.1.4 Querying Details About an ECS ● 4.1.5 Querying Details About ECSs ● 4.2.1 Reinstalling an ECS OS (Using an Image with Cloud-Init Installed) ● 4.2.2 Changing an ECS OS (Using an Image with Cloud-Init Installed) ● 4.2.3 Reinstalling an ECS OS (Using an Image Without Cloud-Init Installed) ● 4.2.4 Changing an ECS OS (Using an Image Without Cloud-Init Installed) ● 4.2.9 Modifying the Specifications of an ECS ● 4.3.4 Modifying ECSs in a Batch ● 4.3.5 Resetting the Passwords for Logging In to ECSs in a Batch ● 4.4.3 Querying the Target Flavors to Which an ECS Flavor Can Be Changed (Discarded) ● 4.11.5 Resetting the Password for Logging In to an ECS with a Few Clicks (Discarded) ● 1.6 Selecting an API Type or Version ● 7.1 Data Structure for Creating ECSs ● 7.3 Data Structure for Query Details About Specifications ● 7.2 Data Structure for Querying Details About ECSs
2018-12-30	<p>This issue is the seventh official release.</p> <ul style="list-style-type: none"> ● Modified 4.1.3 Deleting ECSs. ● Modified 4.3.1 Starting ECSs in a Batch. ● Modified 4.3.2 Restarting ECSs in a Batch. ● Modified 4.3.3 Stopping ECSs in a Batch. ● Added error codes in 9.2 Error Code Description.
2018-11-19	<p>This issue is the sixth official release.</p> <ul style="list-style-type: none"> ● Added 8 Permissions Policies and Supported Actions.

Released On	Description
2018-08-30	This issue is the fifth official release. <ul style="list-style-type: none">• Modified constraints in 5.2.1 Creating an ECS.• Added check rules for the description parameter in 4.1.2 Creating an ECS (Pay-per-Use) and 5.2.1 Creating an ECS.
2018-05-30	This issue is the fourth official release. <ul style="list-style-type: none">• Modified the adminPass field description in 4.1.2 Creating an ECS (Pay-per-Use).
2018-03-30	This issue is the third official release. <ul style="list-style-type: none">• Added 4.6.6 Querying Disk Attachment of an ECS (Discarded).• Added 4.6.7 Querying a Single Disk Attached to an ECS (Discarded).• Added the delete_flag parameter in 4.6.5 Detaching an EVS Disk from an ECS for forcibly detaching a data disk.
2018-01-30	This issue is the second official release. Modified the following content: <ul style="list-style-type: none">• Modified the chargemode and chargingMode field descriptions in 4.1.2 Creating an ECS (Pay-per-Use).
2017-12-31	This issue is the first official release.