

Bare Metal Server

API Reference

Issue 03
Date 2025-10-22



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

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei 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.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:

<https://www.huawei.com/en/psirt/vul-response-process>

For vulnerability information, enterprise customers can visit the following web page:

<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

Contents

- 1 Before You Start..... 1
- 2 API Overview..... 3
- 3 Calling APIs..... 8
 - 3.1 Making an API Request..... 8
 - 3.2 Authentication..... 12
 - 3.3 Response..... 13
- 4 BMS APIs..... 16
 - 4.1 API Instructions..... 16
 - 4.2 API Version Query..... 16
 - 4.2.1 Querying API Versions..... 16
 - 4.2.2 Querying an API Version..... 18
 - 4.3 BMS Lifecycle Management..... 20
 - 4.3.1 Creating BMSs..... 20
 - 4.3.2 Querying Details About a BMS..... 37
 - 4.3.3 Querying Details About BMSs..... 47
 - 4.4 BMS Status Management..... 52
 - 4.4.1 Changing a BMS Name..... 53
 - 4.4.2 Reinstalling a BMS OS..... 63
 - 4.4.3 Starting BMSs..... 68
 - 4.4.4 Restarting BMSs..... 71
 - 4.4.5 Stopping BMSs..... 74
 - 4.5 BMS Metadata Management..... 76
 - 4.5.1 Updating BMS Metadata..... 76
 - 4.6 BMS NIC Management..... 79
 - 4.6.1 Querying NICs Bound to a BMS..... 80
 - 4.7 BMS Disk Management..... 82
 - 4.7.1 Attaching an EVS Disk to a BMS..... 82
 - 4.7.2 Detaching an EVS Disk from a BMS..... 85
 - 4.7.3 Querying EVS Disks Attached to a BMS..... 86
 - 4.8 BMS Password Management..... 88
 - 4.8.1 Querying Whether One-Click Password Reset Is Supported..... 88
 - 4.8.2 Obtaining the Password of a Windows BMS..... 90

4.8.3 Deleting the Password of a Windows BMS.....	91
4.9 BMS Quota Management.....	92
4.9.1 Querying Tenant Quotas.....	93
4.10 BMS Flavor Management.....	95
4.10.1 Querying Flavor Details and Extended Flavor Information.....	95
4.11 Task Management.....	100
4.11.1 Querying Task Statuses.....	100
4.12 BMS SSH Key Management (Native OpenStack APIs).....	105
4.12.1 Querying SSH Key Pairs (Native OpenStack API).....	105
4.12.2 Querying an SSH Key Pair (Native OpenStack API).....	107
4.12.3 Creating or Importing an SSH Key Pair (Native OpenStack API).....	109
4.12.4 Deleting an SSH Key Pair (Native OpenStack API).....	112
5 Examples.....	114
5.1 Example 1: Creating a BMS.....	114
5.2 Example 2: Querying Details About BMSs.....	118
5.3 Example 3: Attaching Disks to a BMS.....	122
6 Public Parameters.....	125
6.1 Status Codes.....	125
6.2 Error Codes.....	126
6.3 Task Responses.....	141
6.3.1 Task ID Response.....	142
6.3.2 Order ID Response.....	143
7 Permissions and Supported Actions.....	145
7.1 Introduction.....	145
7.2 Lifecycle Management.....	147
7.3 Status Management.....	147
7.4 Flavor Query.....	148
7.5 NIC Management.....	148
7.6 Disk Management.....	148
7.7 Metadata Management.....	149
7.8 Tenant Quota Management.....	149
7.9 Password Management.....	149
8 Historical APIs.....	151
8.1 API Instructions (OpenStack Nova APIs).....	151
8.2 API Version Query (OpenStack Nova APIs).....	151
8.2.1 Querying API Versions.....	151
8.2.2 Querying an API Version.....	153
8.3 BMS Lifecycle Management (OpenStack Nova APIs).....	155
8.3.1 Creating a BMS (Deprecated).....	155
8.3.2 Deleting a BMS (Deprecated).....	163
8.3.3 Querying Details About a BMS (Deprecated).....	164

8.3.4 Querying BMSs (Deprecated).....	174
8.3.5 Querying Details About BMSs (Deprecated).....	179
8.4 BMS Status Management (OpenStack Nova APIs).....	192
8.4.1 Changing the Name of a BMS (Deprecated).....	192
8.4.2 Starting a BMS (Deprecated).....	197
8.4.3 Restarting a BMS (Deprecated).....	199
8.4.4 Stopping a BMS (Deprecated).....	200
8.4.5 Obtaining a Remote Login Address (Deprecated).....	202
8.5 BMS Metadata Management (OpenStack Nova APIs).....	204
8.5.1 Querying BMS Metadata (Deprecated).....	204
8.5.2 Updating BMS Metadata (Deprecated).....	206
8.5.3 Modifying Specified BMS Metadata (Deprecated).....	208
8.5.4 Deleting Specified BMS Metadata (Deprecated).....	211
8.6 BMS IP Address Query (OpenStack Nova APIs).....	212
8.6.1 Querying IP Addresses of a BMS (Deprecated).....	212
8.6.2 Querying the Specified IP Address of a BMS (Deprecated).....	214
8.7 BMS Flavor Query (OpenStack Nova APIs).....	216
8.7.1 Querying BMS Flavors (Deprecated).....	216
8.7.2 Querying Details About a BMS Flavor (Deprecated).....	219
8.7.3 Querying Details About extra_specs Parameters of a BMS Flavor (Deprecated).....	222
8.8 BMS NIC Management (OpenStack Nova APIs).....	225
8.8.1 Querying Information About BMS NICs (Deprecated).....	225
8.8.2 Querying Information About a Specified BMS NIC (Deprecated).....	227
8.9 BMS Disk Management (OpenStack Nova APIs).....	230
8.9.1 Querying Information About EVS Disks Attached to a BMS (Deprecated).....	230
8.9.2 Querying Information About an EVS Disk Attached to a BMS (Deprecated).....	232
8.9.3 Attaching an EVS Disk to a BMS (Deprecated).....	234
8.9.4 Detaching an EVS Disk from a BMS (Deprecated).....	236
8.10 BMS 1D Tag Management (OpenStack Nova APIs).....	238
8.10.1 Querying BMS Tags (Deprecated).....	238
8.10.2 Adding Tags for a BMS (Deprecated).....	239
8.10.3 Deleting Tags from a BMS (Deprecated).....	241
8.10.4 Adding a Tag for a BMS (Deprecated).....	243
8.10.5 Checking for Tags of a BMS (Deprecated).....	244
8.10.6 Deleting a Tag from a BMS (Deprecated).....	246
A Appendix.....	248
A.1 Obtaining a Project ID.....	248
A.2 Obtaining an Account ID.....	249

1 Before You Start

Welcome to Bare Metal Server (BMS). A BMS is a physical server that is dedicated for you on the cloud. It provides the excellent computing performance and data security needed for core databases, key application systems, high-performance computing (HPC), and Big Data services. With the high scalability offered by cloud resources, you can apply for and use BMSs flexibly.

This document describes how to use application programming interfaces (APIs) to perform operations on BMSs, such as creating a BMS, attaching disks to a BMS, and start or stop a BMS. For details about all supported operations, see [API Overview](#).

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

BMS provides Representational State Transfer (REST) APIs. You can call APIs using HTTPS. For details about API calling, see [Calling APIs](#).

Endpoints

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

Constraints

- The number of BMSs that you can create is determined by your quota. To view or increase the quota, see section "Adjusting Resource Quotas" in *Bare Metal Server User Guide*.
- For more constraints, see API description.

Concepts

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

- User

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

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

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

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

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

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

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

For details about enterprise projects and about how to obtain enterprise project IDs, see [Enterprise Management User Guide](#).

2 API Overview

API Introduction

BMS APIs (recommended) and Native OpenStack APIs are provided.

Table 2-1 API description

Type	Function	Description
BMS APIs	API version information query	Query the API version of the BMS service.
	Lifecycle management	Create yearly/monthly BMSs and query BMS details.
	Status management	Change the BMS name, reinstall the BMS OS, start, restart, and stop BMSs.
	Flavor management	Query BMS flavor details and extended flavor information, such as the flavor ID, flavor name, number of CPU cores, and boot device.
	NIC management	Query the BMS NIC information, such as the IP address and MAC address of the NIC.
	EVS disk management	Attach EVS disks to or detach EVS disks from a BMS, and query EVS disks attached to a BMS.
	Metadata management	The BMS metadata includes BMS basic information on the cloud platform, such as the BMS ID, hostname, and network information. You can update the BMS metadata.

Type	Function	Description
	Tenant quota management	Query the quotas of all resources of a specified tenant, including used quotas.
	Password management	Query whether one-click password reset is supported. If yes, you can reset the BMS password. Obtain and delete the password of a Windows BMS.
	Task status query	After an asynchronous API is called, for example an API for creating BMSs or attaching or detaching disks, job_id will be returned. You can use job_id to query the task status.
Native OpenStack v2.1 APIs	Lifecycle management	Query BMS details, query BMSs, and query details about BMSs.
	Status management	Start, restart, and stop BMSs.
	Metadata management	The BMS metadata includes BMS basic information on the cloud platform, such as the BMS ID, hostname, and network information. You can query, update, and delete BMS metadata.
	IP address query	Query the private IP address of a BMS, including the IP address version (IPv4 or IPv6) and the specific IP address.
	BMS flavor query	• Querying BMS Flavors: Query all BMS flavors or query specified BMS flavors by filter. • Querying Details About a BMS Flavor: Query details about a BMS flavor based on the flavor ID, such as the flavor name, number of CPU cores, and memory. • Querying Details About extra_specs Parameters of a BMS Flavor: extra_specs parameters are key-value pairs of a BMS flavor. If you want to check whether a BMS flavor supports quick provisioning, call this API.
	BMS NIC query	Query all NICs of a BMS or query details about a NIC based on the NIC ID, such as the IP address and MAC address of the NIC.
	EVS disk management	Query all EVS disks attached to a BMS or query details about an EVS disk attached to a BMS based on the disk ID, such as the mount directory.

Type	Function	Description
	SSH key management	Query SSH key pairs or details about an SSH key pair, and create and delete SSH key pairs.
	1D tag management	Add, delete, modify, and query 1D tags of BMSs.

 NOTE

- When using BMS APIs, you must use the endpoints of the BMS service.
- When using native OpenStack APIs, you must use the endpoints registered by the ECS service.
- In the current version, long HTTP connections cannot be used to call native OpenStack APIs.

BMS API Constraints

Table 2-2 BMS API constraints

Type	API	URI	Constraint
API version query	Querying API Versions	GET /	2,000 requests per minute
	Querying an API Version	GET /{api_version}	2,000 requests per minute
Lifecycle management	Creating a BMS	POST /v1/{project_id}/baremetalservers	50 requests per minute
	Querying Details About a BMS	GET /v1/{project_id}/baremetalservers/detail	500 requests per minute
	Querying Details About BMSs	GET /v1/{project_id}/baremetalservers/{server_id}	1,000 requests per minute
Status management	Changing the Name of a BMS	PUT /v1/{project_id}/baremetalservers/{server_id}	100 requests per minute
	Reinstalling the BMS OS	POST /v1/{project_id}/baremetalservers/{server_id}/reinstallos	50 requests per minute
	Starting a BMS	POST /v1/{project_id}/baremetalservers/action	50 requests per minute

Type	API	URI	Constraint
	Restarting a BMS	POST /v1/{project_id}/baremetalservers/action	50 requests per minute
	Stopping a BMS	POST /v1/{project_id}/baremetalservers/action	50 requests per minute
Flavor management	Querying Flavor Details and Extended Flavor Information	GET /v1/{project_id}/baremetalservers/flavors	500 requests per minute
NIC management	Querying BMS NICs	GET /v1/{project_id}/baremetalservers/{server_id}/os-interface	500 requests per minute
EVS disk management	Attaching an EVS Disk to a BMS	POST /v1/{project_id}/baremetalservers/{server_id}/attachvolume	100 requests per minute
	Detaching an EVS Disk from a BMS	DELETE /v1/{project_id}/baremetalservers/{server_id}/detachvolume/{attachment_id}	100 requests per minute
	Querying Disks Attached to a BMS	GET /v1/{project_id}/baremetalservers/{server_id}/os-volume_attachments	500 requests per minute
Metadata management	Updating the Metadata of a BMS	POST /v1/{project_id}/baremetalservers/{server_id}/metadata	100 requests per minute
Tenant quota management	Querying Tenant Quotas	GET /v1/{project_id}/baremetalservers/limits	500 requests per minute
Password management	Querying Whether One-Click Password Reset Is Supported	GET /v1/{project_id}/baremetalservers/{server_id}/os-resetpwd-flag	500 requests per minute

Type	API	URI	Constraint
	Resetting the BMS Password with a Few Clicks	-	-
	Obtaining the Password of a Windows BMS	GET /v1/{project_id}/baremetalservers/{server_id}/os-server-password	50 requests per minute
	Deleting the Password of a Windows BMS	DELETE /v1/{project_id}/baremetalservers/{server_id}/os-server-password	50 requests per minute
Task management	Querying the Task Status	GET /v1/{project_id}/jobs/{jobId}	2,000 requests per minute

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 [creating an IAM user](#) as an example to demonstrate how to call an API.

Request URI

A request URI is in the following format:

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

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

Table 3-1 URI parameter description

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

IAM is a global service. You can create an IAM user using the endpoint of IAM in any region. For example, to create an IAM user in the **EU-Dublin** region, obtain the endpoint of IAM (**iam.myhuaweicloud.eu**) for this region and the **resource-path** (**/v3.0/OS-USER/users**) in the URI of the API for **creating an IAM user**. Then construct the URI as follows:

```
https://iam.myhuaweicloud.eu/v3.0/OS-USER/users
```

NOTE

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

Request Methods

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

Table 3-2 HTTP methods

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

For example, in the case of the API for **creating an IAM user**, the request method is **POST**. An example request is as follows:

```
POST https://iam.myhuaweicloud.eu/v3.0/OS-USER/users
```

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
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in Obtaining a Project ID .	No This field is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cbaa340f9c0f4

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

The following shows an example request of the API for [creating an IAM user](#) when AK/SK authentication is used:

```
POST https://iam.myhuaweicloud.eu/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****
```

(Optional) Request Body

This part is optional. A request body is generally sent in a structured format (for example, JSON or XML), which is specified by **Content-Type** in the request header. It is used to transfer content other than the request header. If the request body contains full-width characters, these characters must be coded in UTF-8.

The request body varies depending on APIs. Certain APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

The following shows an example request (a request body included) of the API for [creating an IAM user](#). You can learn about request parameters and related description from this example. The bold parameters need to be replaced for a real request.

- **accountid**: account ID of an IAM user
- **username**: name of an IAM user
- **email**: email of an IAM user
- **password**: login password of an IAM user

```
POST https://iam.myhuaweicloud.eu/v3.0/OS-USER/users
Content-Type: application/json
```

```
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****

{
  "user": {
    "domain_id": "accountid",
    "name": "username",
    "password": "*****",
    "email": "email",
    "description": "IAM User Description"
  }
}
```

If all data required for the API request is available, you can send the request to call the API through [curl](#), [Postman](#), or coding.

3.2 Authentication

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

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

Token Authentication

NOTE

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

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

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

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username", // IAM user name
          "password": "*****", // IAM user password
          "domain": {
            "name": "domainname" // Name of the account to which the IAM user belongs
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxx" // Project Name
      }
    }
  }
}
```

```
}  
}
```

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

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

AK/SK Authentication

NOTE

- AK/SK authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token authentication is recommended.
- The AK/SK can either be permanent or temporary. If it is temporary, the **X-Security-Token** field must be included in the request header. The value is the security token of the temporary AK/SK.

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

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

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

NOTE

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

3.3 Response

Status Code

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

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

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

Response Header

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

Figure 3-1 shows the response header fields for the API used to **create an IAM user**. The **X-Subject-Token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

NOTE

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

Figure 3-1 Header fields of the response to the request for creating an IAM user

```
"X-Frame-Options": "SAMEORIGIN",
"X-IAM-ETag-id": "2562365939-d8f6f12921974cb097338ac11fceac8a",
"Transfer-Encoding": "chunked",
"Strict-Transport-Security": "max-age=31536000; includeSubdomains;",
"Server": "api-gateway",
"X-Request-Id": "af2953f2bcc67a42325a69a19e6c32a2",
"X-Content-Type-Options": "nosniff",
"Connection": "keep-alive",
"X-Download-Options": "noopen",
"X-XSS-Protection": "1; mode=block;",
"X-IAM-Trace-Id": "token_ _null_af2953f2bcc67a42325a69a19e6c32a2",
"Date": "Tue, 21 May 2024 09:03:40 GMT",
"Content-Type": "application/json; charset=utf8"
```

(Optional) Response Body

The body of a response is often returned in a structured format (for example, JSON or XML) as specified in the **Content-Type** header field. The response body transfers content except the response header.

The following shows part of the response body for the API used to **create an IAM user**.

```
{
  "user": {
    "id": "c131886aec...",
    "name": "IAMUser",
    "description": "IAM User Description",
    "areacode": "",
    "phone": "",
    "email": "***@***.com",
    "status": null,
    "enabled": true,
    "pwd_status": false,
    "access_mode": "default",
    "is_domain_owner": false,
    "xuser_id": "",
    "xuser_type": "",
    "password_expires_at": null,
    "create_time": "2024-05-21T09:03:41.000000",
    "domain_id": "d78cbac1.....",
    "xdomain_id": "30086000.....",
    "xdomain_type": "",
    "default_project_id": null
  }
}
```

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

```
{  
  "error_msg": "The request message format is invalid.",  
  "error_code": "IMG.0001"  
}
```

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

4 BMS APIs

4.1 API Instructions

- For details about network service APIs, see [Virtual Private Cloud API Reference](#).

4.2 API Version Query

4.2.1 Querying API Versions

Function

This API is used to query all available API versions of the BMS service.

URI

GET /

Request Parameters

None

Example Request

Querying all available API versions
GET https://{BMS Endpoint}/

Response Parameters

Parameter	Type	Description
versions	Array of objects	Specifies API versions of the BMS service. For details, see Table 4-1 .

Table 4-1 versions data structure

Parameter	Type	Description
id	String	Specifies the API version ID.
links	Array of objects	Specifies the API URL. For details, see Table 4-2 .
min_version	String	Specifies the earliest micro API version that is supported.
status	String	Specifies the API version status. • CURRENT: indicates a primary version. • SUPPORTED: indicates an earlier version which is still supported. • DEPRECATED: indicates a deprecated version which may be deleted later.
updated	String	Specifies the release date of an API version. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2018-09-30T00:00:00Z.
version	String	Specifies the latest micro API version that is supported.

Table 4-2 links data structure

Parameter	Type	Description
href	String	Specifies the API URL.
rel	String	Specifies the API URL dependency. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.

Example Response

```
{
  "versions": [
    {
      "id": "v1",
      "links": [
        {
          "href": "http://bms.xxx.com/v1/",
          "rel": "self"
        }
      ],
      "min_version": "",
      "status": "CURRENT",
      "updated": "2018-09-30T00:00:00Z",
    }
  ]
}
```

```
    "version": ""  
  }  
]  
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.2.2 Querying an API Version

Function

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

URI

GET /{api_version}

[Table 4-3](#) lists the parameters.

Table 4-3 Parameter description

Parameter	Mandatory	Description
api_version	Yes	Specifies the API version, for example, v1.

Request Parameters

None

Example Request

Querying the API V1
GET https://{BMS Endpoint}/v1

Response Parameters

Parameter	Type	Description
version	Object	Specifies a specified API version of the BMS service. For details, see Table 4-4 .

Table 4-4 version data structure

Parameter	Type	Description
id	String	Specifies the API version ID.
links	Array of objects	Specifies the API URL. For details, see Table 4-5 .
min_version	String	Specifies the earliest micro API version that is supported.
status	String	Specifies the API version status. ● CURRENT: indicates a primary version. ● SUPPORTED: indicates an earlier version that is still supported. ● DEPRECATED: indicates a deprecated version that may be deleted later.
updated	String	Specifies the release date of an API version. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2018-09-30T00:00:00Z.
version	String	Specifies the latest micro API version that is supported.

Table 4-5 links data structure

Parameter	Type	Description
href	String	Specifies the API URL.
rel	String	Specifies the API URL dependency. The value can be: ● self: resource link that contains the version number. It is used when immediate tracing is required. ● bookmark: resource link that can be stored for a long time.

Example Response

```
{
  "version": {
    "id": "v1",
    "links": [
      {
        "href": "http://bms.xxx.com/v1/",
        "rel": "self"
      }
    ],
    "min_version": "",
    "status": "CURRENT",
    "updated": "2018-09-30T00:00:00Z",
    "version": ""
  }
}
```

Returned Values

Normal values

Return Value	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.3 BMS Lifecycle Management

4.3.1 Creating BMSs

Function

This API is used to create one or multiple BMSs.

Background for Setting the Login Authentication Mode

Logging in to a BMS 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 BMS login authentication.
Method of calling APIs: Use the **key_name** field to specify the key file used for authentication of login to a BMS. For details about how to use the **key_name** field, see [Table 4-7](#).

- Password

If you choose the initial password for authentication in a BMS, you can log in to the BMS using the username and its initial password. The initial password of user **root** is used for authentication in Linux.

Methods of calling APIs:

- Method 1 (recommended): Use the **adminPass** field to specify the initial login password of the specified administrator account. For details about how to use the **adminPass** field, see [Table 4-7](#).

 NOTE

For a Linux BMS with Cloud-Init installed, if field **user_data** is specified, field **adminPass** is invalid.

- Method 2:
 - For a Linux BMS with Cloud-Init installed, use the **user_data** field to inject data. For details, see [Table 4-7](#).

 NOTE

Public images contain Cloud-Init or Cloudbase-Init by default. For private images, you need to check whether Cloud-Init or Cloudbase-Init is installed.

Constraints

- File injection is not supported.
- Currently, only yearly/monthly BMSs can be created.

URI

POST /v1/{project_id}/baremetalservers

[Table 4-6](#) lists the parameters.

Table 4-6 Parameter description

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

Request Parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies BMS information. For details, see Table 4-7 .

Table 4-7 server data structure

Parameter	Mandatory	Type	Description
imageRef	Yes	String	Specifies the image ID or image resource URL used for creating the BMS. The ID is in the format of a Universally Unique Identifier (UUID). You can obtain the image ID from the IMS console or by following the instructions in Querying Images in <i>Image Management Service API Reference</i>. When using an API to query images, you can add the field ?virtual_env_type=ironic to filter BMS images.
flavorRef	Yes	String	Specifies the flavor ID of the BMS. The format is physical.x.x. You can obtain the flavor ID from the BMS console or using the Querying Flavor Details and Extended Flavor Information API. NOTE For details about the images supported by different BMS flavors, see OSs Supported by Different Types of BMSs. Only DeC tenants can apply for BMSs using the physical.x.x.hba flavor, and only DESS disks can be attached to BMSs using this flavor.
name	Yes	String	Specifies the BMS name. Value range: • The value can contain 1 to 63 characters, consisting of letters, digits, underscores (_), hyphens (-), and periods (.). • If more than one BMS is to be created, suffixes similar to -0000 will be automatically added to the end of the BMS names during the BMS creation. In this case, the BMS name contains 1 to 58 characters.
metadata	Yes	Object	Specifies the BMS metadata. The maximum size for both the metadata key and value is 255 characters. For details, see Table 4-8.

Parameter	Mandatory	Type	Description
user_data	No	String	Specifies the user data to be injected during the BMS creation. Text can be injected. Constraints: • The content to be injected must be encoded with base64. The maximum size of the content to be injected (before encoding) is 32 KB. • If key_name is not specified, the password of user root for logging in to the BMS will be injected by default. • This parameter is mandatory when you create a Linux BMS using the password authentication mode. Its value is the initial user root password. Password complexity requirements: • Contains 8 to 26 characters. • Contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters <code>!@\$%^_-=+[{ }],./?</code> An example is as follows: <pre>#!/bin/bash echo 'root:\$6\$V6azyeLwcD3CHlpY \$BN3VVq18fmCkj66B4zdHLWevqcxlgi' chpasswd -e</pre> where, \$6\$V6azyeLwcD3CHlpY \$BN3VVq18fmCkj66B4zdHLWevqcxlgi is the ciphertext password, which can be generated as follows: 1. Generate an encrypted salt value. <pre>[root@test linux]# python -c "import crypt, getpass, pwd;print crypt.mksalt()" \$6\$V6azyeLwcD3CHlpY</pre> 2. Generate a ciphertext password based on the salt value. <pre>[root@test linux]# python -c "import crypt, getpass, pwd;print crypt.crypt('Cloud.1234','\'\$6\ \$V6azyeLwcD3CHlpY')" \$6\$V6azyeLwcD3CHlpY \$BN3VVq18fmCkj66B4zdHLWevqcxlgi</pre> NOTE Data injection is not supported for BMSs that use a Linux image and the password login mode. For details about how to inject user data, see Injecting User Data into BMSs.

Parameter	Mandatory	Type	Description
adminPassword	No	String	Specifies the initial login password of the administrator account for logging in to a BMS. The Linux administrator is root. Password complexity requirements: • Contains 8 to 26 characters. • Contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^&_+=+[{ }],./? • (Linux OSs) Cannot contain the username or the username in reverse.
key_name	No	String	Specifies the name of a key pair. This is an extended attribute. To log in to a BMS using an SSH key pair, set the value to the name of an existing private key. You can create a key pair using the Creating or Importing an SSH Key Pair (Native OpenStack API) API, or query existing key pairs using the Querying SSH Key Pairs (Native OpenStack API) API. Constraints: • If both key_name and user_data are specified, user_data only injects user data. • When a Windows image is used to create a BMS, this parameter cannot be left blank because a Windows image supports only login using a key pair. • To log in to a Windows BMS, decrypt the key pair to obtain a password and use it to remotely log in to the BMS. For details, see Logging In to a BMS Remotely Using MSTSC.
security_groups	No	Array of objects	Specifies security groups of the BMS. For details, see Table 4-9 .
nics	Yes	Array of objects	Specifies NICs of the BMS. For details, see Table 4-10. Constraints: A maximum of two NICs can be attached to a BMS. The first will be used as the primary NIC. If multiple NICs are specified, ensure that all NICs belong to the same VPC.
availability_zone	Yes	String	Specifies the name of the AZ where the BMS is.

Parameter	Mandatory	Type	Description
vpcid	Yes	String	Specifies the ID of the VPC where the BMS is. The value is in UUID format. You can obtain the VPC ID from the network console or by following the instructions in Querying VPC Details of <i>Virtual Private Cloud API Reference</i> .
publicip	No	Object	Specifies the EIP information of the BMS. Possible values include: • Do not use (This parameter is unavailable.) • Automatically assign: Assign a new EIP. • Specify: Specify an EIP that has been created. For details, see Table 4-11 .
count	No	Integer	Specifies the number of BMSs to be created. Constraints: • If this parameter is not specified, the default value is 1. • If the quota is sufficient, the maximum value is 100.
root_volume	No	Object	Specifies system disk details of the BMS. For details, see Table 4-12 . Constraints: If the flavor supports quick provisioning, this parameter is mandatory. Otherwise, this parameter is not required. For how to check whether a flavor supports quick provisioning, see Querying Flavor Details and Extended Flavor Information .
data_volumes	No	Array of objects	Specifies data disk details of the BMS. Each data structure represents a data disk to be created. For details, see Table 4-13 . Constraints: A maximum of 60 EVS disks (including the system disk and data disks) can be attached to a BMS.
extendparam	Yes	Object	Specifies the supplementary for creating the BMS. For details, see Table 4-14 .
schedulerHints	No	Object	Specifies scheduling information of the BMS. This parameter is mandatory for creating a BMS in a DeC. For details, see Table 4-15 .

Parameter	Mandatory	Type	Description
server_tags	No	List <resource_tag>	Specifies tags of the BMS. For details, see Table 4-16 . NOTE A maximum of 10 tags can be added for a BMS. There has been a system tag __type_baremetal by default. So, you can add a maximum of nine tags.

Table 4-8 metadata data structure

Parameter	Mandatory	Type	Description
op_svc_userid	Yes	String	Specifies the user ID. You can obtain the user ID from My Credential on the management console.
BYOL	No	String	Specifies whether a license is provided. The value can be true or false .
admin_pass	No	String	Specifies the initial password of the administrator account for logging in to a BMS. For Linux, the administrator account is root . Password complexity requirements: - Contains 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, the username in reverse, or more than two consecutive characters in the username.
agency_name	No	String	Specifies the IAM agency name. An agency provides a temporary security credential for accessing a BMS. The agency is created by the tenant administrator on the IAM console. NOTE To obtain and update an agency, perform the following steps: - You can obtain a valid agency name by using the Listing Agencies API provided by IAM. - Use the Updating the Metadata of a BMS API to change the value of field agency_name in the metadata to the new agency name.

Table 4-9 security_groups data structure

Parameter	Mandatory	Type	Description
id	No	String	Specifies a security group ID, which takes effect for all NICs configured for the BMS. • If this parameter is not specified, the default security group is bound to the BMS. • If this parameter is required (in UUID format), use the ID of an existing security group. For details about how to obtain existing security groups, see Querying Security Groups in <i>Virtual Private Cloud API Reference</i>.

Table 4-10 nics data structure

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	Specifies the subnet information of a BMS NIC. The value must be the ID of the subnet (network_id) created in the VPC specified by vpcid and in the format of UUID. You can obtain the subnet ID (network_id) from the VPC console or by following the instructions in Querying Subnets in <i>Virtual Private Cloud API Reference</i>.
ip_address	No	String	Specifies the IPv4 address of a BMS NIC. Constraints: • If this parameter is left blank or set to "", an unused IP address in the subnet of this network 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. • The IP address cannot be specified when you create BMSs in a batch.

Table 4-11 publicip data structure

Parameter	Mandatory	Type	Description
id	No	String	Specifies the ID of an existing EIP assigned to the BMS. The value is in UUID format. You can obtain the EIP ID from the network console or by following the instructions in Querying EIPs in <i>Elastic IP API Reference</i> . Constraints: - Only EIPs in the DOWN state can be assigned. - Existing EIPs cannot be used for creating BMSs in a batch. That is, this parameter is invalid in such a case.
eip	No	Object	Specifies the configuration for creating an EIP that will be automatically assigned to the BMS. For details, see Table 4-17 .

 NOTE

You can configure either but not both of **id** and **eip** in the **publicip** field.

Table 4-12 root_volume data structure

Parameter	Mandatory	Type	Description
volumetype	Yes	String	Specifies the BMS system disk type. The disk type must match the available disk type. - GPSSD: general-purpose SSD disk type - SAS: high I/O disk type - SSD: ultra-high I/O disk type
size	Yes	Integer	Specifies the system disk size (GB). The value ranges from 40 to 1024 . Constraints: The system disk size must be greater than or equal to the minimum system disk size of the image (min_disk attribute).
cluster_id	No	String	Specifies the ID of the storage pool that the BMS system disk belongs to. NOTE This parameter is required when Dedicated Distributed Storage Service (DSS) is used.

Parameter	Mandatory	Type	Description
cluster_type	No	String	Specifies the storage type of the BMS system disk. An example value is DSS , which indicates the DSS service. NOTE This parameter is required when DSS is used.

Table 4-13 data_volumes data structure

Parameter	Mandatory	Type	Description
volumetype	Yes	String	Specifies the BMS data disk type. The disk type must match the available disk type. - GPSSD: general-purpose SSD disk type - SAS: high I/O disk type - SSD: ultra-high I/O disk type
size	Yes	Integer	Specifies the data disk size (GB). The value ranges from 10 to 32768 .
shareable	No	Boolean	Specifies whether the disk is shareable. true: shared EVS disk false: common EVS disk The default value is false .
cluster_id	No	String	Specifies the ID of the storage pool to which BMS data disks belong. NOTE This parameter is required when Dedicated Distributed Storage Service (DSS) is used.
cluster_type	No	String	Specifies the storage type of BMS data disks. An example value is DSS , which indicates the DSS service. NOTE This parameter is required when DSS is used.

Table 4-14 extendparam data structure for creating BMSs

Parameter	Mandatory	Type	Description
charging Mode	No	String	Specifies the billing mode. Value range: prePaid (default): yearly/monthly billing

Parameter	Mandatory	Type	Description
regionID	No	String	Specifies the ID of the region where the BMS resides.
periodType	No	String	Specifies the subscription period. Value range: • month: indicates that the subscription unit is month. • year: indicates that the subscription unit is year. NOTE This parameter is valid and mandatory if chargingMode is set to prePaid .
periodNum	No	Integer	Specifies the number of subscription periods. Value range: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value is 1. NOTE This parameter is valid and mandatory if chargingMode is set to prePaid .
isAutoRenew	No	String	Specifies whether auto-renewal is enabled. • true: indicates that auto-renewal is enabled. • false: indicates that auto-renewal is disabled. NOTE This parameter is valid when chargingMode is set to prePaid . If this parameter is not specified or is left blank, auto-renewal is disabled.
isAutoPay	No	String	Specifies whether the order is automatically or manually paid. • true: The order will be automatically paid. • false: You must manually pay the order. NOTE This parameter is valid when chargingMode is set to prePaid . If this parameter is not specified or is left blank, manual payment is required.

Table 4-15 schedulerHints data structure

Parameter	Mandatory	Type	Description
dec_baremetal	No	String	Specifies whether to create the BMS in a DeC. The value can be share or dedicate . Constraints: - If this parameter is not specified, the default value is share. - To create the BMS in a DeC, set this parameter to dedicate.

Table 4-16 server_tags data structure

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. - It contains a maximum of 36 Unicode characters and cannot be empty. - It cannot contain ASCII characters (0–31) or special characters =*<>\ / - The tag key of a BMS must be unique.
value	No	String	Specifies the tag value. - Each value contains a maximum of 43 Unicode characters and can be an empty string. - It cannot contain ASCII characters (0–31) or special characters =*<>\ /

Table 4-17 eip data structure

Parameter	Mandatory	Type	Description
iptype	Yes	String	Specifies the EIP type. Enumerated values: 5_bgp and 5_sbgp For details, see the publicip field in Assigning an EIP in <i>Elastic IP API Reference</i> .
bandwidth	Yes	Object	Specifies the EIP bandwidth. For details, see Table 4-18 .
extendparam	Yes	Object	Provides additional information about the EIP. For details, see Table 4-19 .

Table 4-18 bandwidth data structure

Parameter	Mandatory	Type	Description
name	No	String	Specifies the bandwidth name.
sharetype	Yes	String	Specifies the bandwidth sharing type. Value PER indicates dedicated bandwidth and WHOLE indicates shared bandwidth.
id	No	String	Specifies the shared bandwidth ID. You can specify an existing shared bandwidth when applying for an EIP with a WHOLE bandwidth. NOTE This parameter is mandatory when sharetype is set to WHOLE .
size	Yes	Integer	- The value ranges from 5 Mbit/s to 2000 Mbit/s by default. (The specific range may vary depending on the configuration in each region. You can view the bandwidth range of each region on the management console.) - Specifies the bandwidth (Mbit/s). The minimum shared bandwidth is 5 Mbit/s by default. NOTE - If a decimal fraction (for example 10.2) or a character string (for example 10) is specified, the specified value will be automatically converted to an integer. If the bandwidth is less than 300 Mbit/s, the step is 1 Mbit/s. If the bandwidth is from 300 Mbit/s to 1000 Mbit/s, the step is 50 Mbit/s. If the bandwidth is from 1000 Mbit/s to 2000 Mbit/s, the step is 1000 Mbit/s. - This parameter is mandatory when sharetype is set to PER and is optional when sharetype is set to WHOLE with an ID specified.
chargemode	No	String	Specifies the bandwidth billing mode. The value can be traffic or bandwidth . - If this field is not specified, the BMS is billed by bandwidth. - If the field value is empty, the BMS is billed by bandwidth.

Table 4-19 extendparam data structure for assigning an EIP

Parameter	Mandatory	Type	Description
chargingMode	Yes	String	Specifies the billing mode of an EIP. If bandwidth is charged by bandwidth, both prePaid and postPaid will be available for EIP. If bandwidth is charged by traffic, only postPaid will be available for EIP. Value range: • prePaid: yearly/monthly billing mode. • postPaid: indicates the pay-per-use billing mode.

Example Request

- Creating a BMS (image ID: 766a545a-02e1-433e-b1d1-733b5dc95e94; flavor: physical.s3.large; billing: yearly/monthly; remote login: password)

```
POST https://{BMS Endpoint}/v1/{project_id}/baremetalservers
{
  "server": {
    "availability_zone": "az1",
    "name": "bms-75c7",
    "imageRef": "766a545a-02e1-433e-b1d1-733b5dc95e94",
    "flavorRef": "physical.s3.large",
    "data_volumes": [
      {
        "volumetype": "SAS",
        "size": 10,
        "shareable": false
      }
    ],
    "vpcid": "8df83bf2-fd2e-4ee0-9692-c0b7736513fb",
    "nics": [
      {
        "subnet_id": "c6bb8788-4fd1-4a8f-adab-7eba3bed8616",
        "ip_address": ""
      }
    ],
    "count": 1,
    "schedulerHints": {
      "dec_baremetal": "share"
    },
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,
      "isAutoRenew": "true",
      "isAutoPay": "true",
      "regionID": "region01"
    },
    "metadata": {
      "op_svc_userid": "59781460e9e54886a7d03df7d3f3fc81",
      "BYOL": "false",
      "admin_pass": "$ADMIN_PASS"
    },
    "publicip": {
      "id": "1b036bce-26b4-465c-acea-99fdbb0573a3"
    }
  }
}
```

```
    },
    "user_data": "$USER_DATA",
    "security_groups": [
      {
        "id": "993f2916-5bad-4f24-bc3c-db539f1a6d19"
      }
    ]
  }
}
```

- Creating a Linux BMS in AZ1 (name: bms-3b91; image ID: 766a545a-02e1-433e-b1d1-733b5dc95e94; flavor: physical.o2.large; billing: yearly/monthly; remote login: key pair)

```
{
  "server": {
    "availability_zone": "az1",
    "name": "bms-3b91",
    "imageRef": "766a545a-02e1-433e-b1d1-733b5dc95e94",
    "flavorRef": "physical.o2.large",
    "data_volumes": [
      {
        "volumetype": "SAS",
        "size": 10,
        "shareable": true
      }
    ],
    "vpcid": "8df83bf2-fd2e-4ee0-9692-c0b7736513fb",
    "nics": [
      {
        "subnet_id": "c6bb8788-4fd1-4a8f-adab-7eba3bed8616",
        "ip_address": ""
      },
      {
        "subnet_id": "2b8bcf33-33f2-4a63-aa9c-b1e17ff0ed7f",
        "ip_address": ""
      }
    ],
    "count": 1,
    "schedulerHints": {
      "dec_baremetal": "share"
    },
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,
      "isAutoRenew": "true",
      "isAutoPay": "true",
      "regionID": "region01"
    },
    "metadata": {
      "op_svc_userid": "59781460e9e54886a7d03df7d3f3fc81",
      "BYOL": "false"
    },
    "publicip": {
      "eip": {
        "iptype": "5_bpg",
        "bandwidth": {
          "chargemode": "",
          "name": "bms-3b91-bandwidth",
          "size": 1,
          "sharetype": "PER"
        },
        "extendparam": {
          "chargingMode": "prePaid"
        }
      }
    },
    "security_groups": [
      {

```

```
        "id": "993f2916-5bad-4f24-bc3c-db539f1a6d19"
      }
    ],
    "key_name": "$key_name"
  }
}
```

The following is an example of **user_data** generated by Java:

```
String userData_org = "#!/bin/bash \r\n echo  'root:xxxxx' | chpasswd ";
byte[] userData_byte = userData_org.getBytes();
String userData = new BASE64Encoder().encode(userData_byte);
```

- Creating a Windows BMS in AZ1 (name: bms-6a1e; image ID: fd04041a-5f35-45ae-a27c-bf30f921c6b5; billing: yearly/monthly; flavor: physical.s3.large; remote login: key pair)

```
{
  "server": {
    "availability_zone": "az1",
    "name": "bms-6a1e",
    "imageRef": "fd04041a-5f35-45ae-a27c-bf30f921c6b5",
    "flavorRef": "physical.s3.large",
    "data_volumes": [
      {
        "volumetype": "SAS",
        "size": 10,
        "shareable": false
      }
    ],
    "vpcid": "0adc2847-c160-4263-be1f-e03d36e93e32",
    "nics": [
      {
        "subnet_id": "4f1cf6cc-4cc3-41d3-8192-625293dcccce0",
        "ip_address": ""
      }
    ],
    "count": 1,
    "schedulerHints": {
      "dec_baremetal": "share"
    },
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,
      "isAutoRenew": "true",
      "isAutoPay": "true",
      "regionID": "region1"
    },
    "metadata": {
      "op_svc_userid": "8a74baed155b47e8b59fbca853d6cb48",
      "BYOL": "false"
    },
    "publicip": {
      "id": "db9c4f39-fabd-4cc2-8685-82e7715d099f"
    },
    "user_data": "$USER_DATA",
    "security_groups": [
      {
        "id": "33cf5e40-0b65-4e03-867d-4c67947c49be"
      }
    ],
    "key_name": "$key_name"
  }
}
```

Response Parameters

Table 4-20 Normal response

Parameter	Type	Description
order_id	String	Specifies the order ID returned after an order is submitted. You can query the order processing progress based on the ID. To query the order processing progress, log in to the management console and choose Billing > My Orders .
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

Table 4-21 Abnormal response

Parameter	Type	Description
error	Dictionary data structure [1]	Specifies the error returned when a task submission encounters an exception. For details, see Table 4-22 .

Table 4-22 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

```
{  "order_id": "CS2009141523OQSEQ",  "job_id": "ff808081748b760c01748b7f80370003"}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.3.2 Querying Details About a BMS

Function

This API is used to query details about a BMS.

The information that can be queried includes the BMS billing mode and whether the BMS is frozen.

URI

GET /v1/{project_id}/baremetalservers/{server_id}

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

Table 4-23 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying details about a BMS (ID: 53206ed0-56de-4d6b-b7ee-ffc62ca26f43)
GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/
53206ed0-56de-4d6b-b7ee-ffc62ca26f43

Response Parameters

Parameter	Type	Description
server	Object	Specifies the BMS information. For details, see Table 4-24 .

Table 4-24 server data structure

Parameter	Type	Description
id	String	Specifies the BMS ID in UUID format.
user_id	String	Specifies the ID of the user that created the BMS. The value is in UUID format.
name	String	Specifies the BMS name.
created	String	Specifies the time when the BMS was created. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T03:30:52Z.
updated	String	Specifies the time when the BMS was updated. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T04:30:52Z.
tenant_id	String	Specifies the ID of the tenant owning the BMS. The value is in UUID format. This parameter specifies the same meaning as project_id .
hostId	String	Specifies the host ID of the BMS.
addresses	Map<String,Array of address objects>	Specifies the VPC where the BMS is. key : indicates the VPC ID. value indicates the VPC details.
key_name	String	Specifies the key pair that is used to authenticate the BMS.
image	Object	Specifies the BMS image. For details, see Table 4-26 .
flavor	Object	Specifies the BMS flavor. For details, see Table 4-27 .
security_groups	Array of objects	Specifies security groups of the BMS. For details, see Table 4-28 .

Parameter	Type	Description
accessIPv4	String	This is a reserved attribute.
accessIPv6	String	This is a reserved attribute.
status	String	Specifies the BMS status. Value range: • ACTIVE: Running, Stopping, Deleting • BUILD: Creating • ERROR: Faulty • HARD_REBOOT: Forcibly Restarting • REBOOT: Restarting • SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen
progress	Integer	This is a reserved attribute.
config_drive	String	Specifies whether to configure a config drive for the BMS. The value can be True or an empty string.
metadata	object	Specifies the BMS metadata. For details, see Table 4-32. NOTE Metadata includes system default fields and the fields set by users.
OS-EXT-STS:task_state	String	Specifies the task status of the BMS. This is an extended attribute. Examples: • rebooting: The BMS is being restarted. • reboot_started: The BMS is normally restarted. • reboot_started_hard: The BMS is forcibly restarted. • powering-off: The BMS is being powered off. • powering-on: The BMS is being powered on. • rebuilding: The BMS is being rebuilt. • scheduling: The BMS is being scheduled. • deleting: The BMS is being deleted.

Parameter	Type	Description
OS-EXT-STS:vm_state	String	Specifies the BMS status. This is an extended attribute. Examples: • active: The BMS is running. • stopped: The BMS is stopped.
OS-EXT-SRV-ATTR:host	String	Specifies the hostname of the BMS. This is an extended attribute.
OS-EXT-SRV-ATTR:instance_name	String	Specifies the BMS alias. This is an extended attribute.
OS-EXT-STS:power_state	Integer	Specifies the power status of the BMS. This is an extended attribute. Examples: • 0: indicates NO STATE. • 1: indicates RUNNING. • 4: indicates SHUTDOWN.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the hostname of the hypervisor.
OS-EXT-AZ:availability_zone	String	Specifies the AZ where the BMS is.
OS-DCF:diskConfig	String	Specifies the disk configuration. This is an extended attribute. The value can be: • MANUAL: The BMS is created using the partitioning scheme and file system in the image. If the target flavor has a large disk, the remaining disk space cannot be partitioned. • AUTO: The BMS is created using a single partition with the same size as the disk of the target flavor. The file system can automatically adapt to the entire partition.
fault	Object	Specifies the cause of a BMS fault. For details, see Table 4-29 .
OS-SRV-USG:launched_at	String	Specifies the time when the BMS was started. The timestamp format is ISO 8601, for example, 2019-05-22T03:23:59.000000 .

Parameter	Type	Description
OS-SRV-USG:terminated_at	String	Specifies the time when the BMS was deleted. The timestamp format is ISO 8601, for example, 2019-05-22T04:23:59.000000 .
os-extended-volumes:volumes_attached	Array of objects	Specifies the disks attached to the BMS. For details, see Table 4-30 .
description	String	Specifies description about the BMS.
host_status	String	Specifies the status of the BMS host. • 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 the maintenance state. • Empty string: There is no BMS host information.
OS-EXT-SRV-ATTR:hostname	String	Specifies the host name of the BMS.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies reserved IDs for multiple BMSs created in a batch. For BMSs created in a batch, they have the same reservation_id .
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the startup sequence for BMSs created in a batch.
OS-EXT-SRV-ATTR:kernel_id	String	Specifies the UUID of the kernel image when the 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 when the AMI image is used. In other scenarios, leave this parameter blank.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the BMS system disk, for example, /dev/sda .

Parameter	Type	Description
OS-EXT-SRV-ATTR:user_data	String	Specifies the user_data specified during BMS creation. The value is encoded using Base64 or an empty string.
locked	Boolean	Specifies whether the BMS is locked. • true: The BMS is locked. • false: The BMS is not locked.
tags	Array of strings	Specifies tags of the BMS.
os:scheduler_hints	Object	Specifies scheduling information of the BMS. For details, see Table 4-31 .
enterprise_project_id	String	Specifies the ID of the enterprise project that the BMS belongs to.
sys_tags	Array of objects	Specifies system tags of the BMS. For details, see Table 4-33 .

Table 4-25 address data structure

Parameter	Type	Description
version	String	Specifies the IP address version. • 4: indicates IPv4. • 6: indicates IPv6.
addr	String	Specifies the IP address.
OS-EXT-IPS:type	String	Specifies the IP address type. • fixed: indicates the private IP address. • floating: indicates the EIP.
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 4-26 image data structure

Parameter	Type	Description
id	String	Specifies the image ID in UUID format.

Table 4-27 flavor data structure

Parameter	Type	Description
id	String	Specifies the BMS flavor ID.
name	String	Specifies the BMS flavor name.
disk	String	Specifies the system disk size in the BMS flavor. 0 indicates the size is not limited.
vcpus	String	Specifies the number of CPU cores in the BMS flavor.
ram	String	Specifies the memory size (MB) in the BMS flavor.

Table 4-28 security_groups data structure

Parameter	Type	Description
name	String	Specifies the security group name.
id	String	Specifies the security group ID.

Table 4-29 fault data structure

Parameter	Type	Description
message	String	Specifies the fault information.
code	Integer	Specifies the fault code.
details	String	Specifies the fault details.
created	String	Specifies the time when the fault occurred. The time is in ISO 8601 format.

Table 4-30 os-extended-volumes:volumes_attached data structure

Parameter	Type	Description
id	String	Specifies the disk ID in UUID format.

Parameter	Type	Description
delete_on_termination	String	Specifies whether to delete the disk when deleting the BMS. • true: Yes • false: No
bootIndex	String	Specifies whether it is a boot disk (system disk). • 0 indicates a system disk. • Non-0 indicates a data disk.
device	String	Specifies the device name of the disk, for example, /dev/sdb .

Table 4-31 os:scheduler_hints data structure

Parameter	Type	Description
dec_baremetal	Array of strings	Specifies whether to create the BMS in a DeC. The value can be share or dedicate .

Table 4-32 metadata data structure

Parameter	Type	Description
chargingMode	String	Specifies the billing type of the BMS. 1 : yearly/monthly billing mode (pre-paid)
metering.order_id	String	Specifies the order ID for a yearly/monthly BMS.
metering.product_id	String	Specifies the product ID for a yearly/monthly BMS.
vpc_id	String	Specifies the ID of the VPC where the BMS is.
metering.image_id	String	Specifies the image ID of the BMS.

Parameter	Type	Description
metering.imagetype	String	Specifies the image type. The following types are supported: - Public image: The value is gold. - Private image: The value is private. - Shared image: The value is shared.
baremetalPortIDList	String	Specifies NICs of the BMS.
metering.resourcesp eccode	String	Specifies the flavor code of the BMS in <i>{Flavor ID}.{os_type}</i> format, for example, physical.o2.medium.linux .
metering.resourcety pe	String	Specifies the resource type of the BMS. The value is __type_baremetal .
image_name	String	Specifies the image name of the BMS.
op_svc_userid	String	Specifies the user ID. You can obtain the user ID from My Credential on the management console.
os_type	String	Specifies the OS type. The value can be Linux or Windows .
__bms_support_evs	String	Specifies whether the BMS can use EVS disks.
os_bit	String	Specifies the number of bits in the OS: 32 or 64 .

Table 4-33 sys_tags data structure

Parameter	Type	Description
key	String	Specifies the key of a system tag. The value is _sys_enterprise_project_id , indicating an enterprise project ID.
value	String	Specifies the system tag value.

Example Response

```
{
  "server": {
```

```
"id": "53206ed0-56de-4d6b-b7ee-ffc62ca26f43",
"name": "bms-test",
"addresses": {
  "5849fdf1-9d79-4589-80c2-fe557990c417": [
    {
      "version": "4",
      "addr": "192.168.1.216",
      "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:25:56:c3",
      "OS-EXT-IPS:port_id": "9e62503b-094e-4c6e-bc95-f10bbfb455d5",
      "OS-EXT-IPS:type": "fixed"
    },
    {
      "version": "4",
      "addr": "10.154.72.77",
      "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:25:56:c3",
      "OS-EXT-IPS:port_id": "9e62503b-094e-4c6e-bc95-f10bbfb455d5",
      "OS-EXT-IPS:type": "floating"
    }
  ]
},
"flavor": {
  "disk": "9309",
  "vcpus": "32",
  "ram": "193047",
  "id": "physical.o2.medium",
  "name": "physical.o2.medium"
},
"accessIPv4": "",
"accessIPv6": "",
"status": "ACTIVE",
"progress": 0,
"hostId": "cd243addb5d2c64e89218180b7a3ed95abe6882e81c337cc563137df",
"updated": "2018-09-18T09:29:44Z",
"created": "2018-09-18T02:43:26Z",
"metadata": {
  "metering.order_id": "CS1809181040HCFC2",
  "metering.image_id": "8589958c-6bc7-40c6-b81b-3fe4cd2e3d85",
  "baremetalPortIDList": "[9e62503b-094e-4c6e-bc95-f10bbfb455d5]",
  "metering.resourcespeccode": "physical.o2.medium.linux",

  "image_name": "redhat_7_2",
  "op_svc_userid": "745973c535ec4d4caba86f6f9419ff6d",
  "metering.resourcetype": "__type_baremetal",

  "metering.product_id": "00301-74069-0--0",
  "os_bit": "64",
  "vpc_id": "5849fdf1-9d79-4589-80c2-fe557990c417",
  "os_type": "Linux",
  "chargingMode": "1"
},
"tags": [
  "__type_baremetal"
],
"description": "bms-6e18",
"locked": false,
"image": {
  "id": "8589958c-6bc7-40c6-b81b-3fe4cd2e3d85"
},
"config_drive": "",
"tenant_id": "bbf1946d374b44a0a2a95533562ba954",
"user_id": "3fc5ab2b0c544979abcaafd86edd80e6",
"OS-EXT-STS:power_state": 1,
"OS-EXT-STS:vm_state": "active",
"OS-EXT-SRV-ATTR:host": "bms.dc1",
"OS-EXT-SRV-ATTR:instance_name": "instance-00154f94",
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova002@2",
"OS-DCF:diskConfig": "MANUAL",
"OS-EXT-AZ:availability_zone": "az-dc-1",
"os:scheduler_hints": {
```

```
    },
    "OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
    "OS-EXT-SRV-ATTR:ramdisk_id": "",
    "enterprise_project_id": "2c7b6d77-9013-4d74-a221-2f612fc56372",
    "OS-EXT-SRV-ATTR:user_data": "$USER_DATA",
    "OS-SRV-USG:launched_at": "2018-09-18T02:46:07.954587",
    "OS-EXT-SRV-ATTR:kernel_id": "",
    "OS-EXT-SRV-ATTR:launch_index": 0,
    "host_status": "UP",
    "OS-EXT-SRV-ATTR:reservation_id": "r-08tuyo8v",
    "OS-EXT-SRV-ATTR:hostname": "bms-6e18",
    "sys_tags": [
      {
        "key": "_sys_enterprise_project_id",
        "value": "2c7b6d77-9013-4d74-a221-2f612fc56372"
      }
    ]
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.3.3 Querying Details About BMSs

Function

This API is used to query BMSs by filters and display details about the BMSs.

The information that can be queried includes the BMS billing mode and whether the BMS is frozen.

URI

GET /v1/{project_id}/baremetalservers/detail

[Table 4-34](#) lists the parameters.

Table 4-34 Parameter description

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

Table 4-35 Parameters for querying BMS details

Parameter	Mandator y	Type	Description
flavor	No	String	Specifies the BMS flavor ID.
name	No	String	Specifies the BMS name.
status	No	String	Specifies the BMS status. Value range: • ACTIVE: Running, Stopping, Deleting • BUILD: Creating • ERROR: Faulty • HARD_REBOOT: Forcibly Restarting • REBOOT: Restarting • SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen
limit	No	Integer	Specifies the number of BMSs displayed on each page. The default value is 25 and the maximum value is 1000 .

Parameter	Mandatory	Type	Description
offset	No	Integer	This API is a pagination query API. offset indicates the page number (the start page number is 1). The returned value contains the number of BMSs and details about the BMSs. • If offset is specified: BMSs are displayed by page based on the limit value. By default, the limit value is 1000. The BMSs and total number of BMSs on the offset page are displayed. The maximum number of BMSs is the value of limit. If the number of BMSs is less than the value of limit, the actual number of BMSs is displayed. • If offset is not specified: – If limit is specified: The BMSs and total number of BMSs are displayed. The maximum number of BMSs is the value of limit. If the number of BMSs is less than the value of limit, the actual number of BMSs is displayed. – If limit is not specified: 25 BMSs are displayed on each page. Details about the BMSs on the first page are displayed. If the number of BMSs is less than 25, the actual number of BMSs is displayed.
tags	No	String	Specifies the BMS tag. The value can be: • __type_baremetal: internal tag of the system • Other custom tags
reservation_id	No	String	Specifies the reserved ID, which can be used to query BMSs created in a batch.
detail	No	String	Specifies the level for details about BMS query results. A higher level indicates more details. Available levels include 4, 3, 2, and 1. The default level is 4.

Request Parameters

None

Example Request

Querying BMSs bound to all enterprise projects of the user

```
GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/detail?offset=1&limit=2&enterprise_project_id=all_granted_eps
```

Response Parameters

Parameter	Type	Description
servers	List data structure	Specifies details about BMSs. For details, see Querying Details About a BMS . The returned details vary depending on the query level.
count	Integer	Specifies the number of BMSs that match the filters.

Example Response

```
{
  "count": 2,
  "servers": [
    {
      "id": "b544be62-1b3b-4982-ad98-572b002ac23b",
      "name": "bms-test1",
      "addresses": {
        "5849fdf1-9d79-4589-80c2-fe557990c417": [
          {
            "version": "4",
            "addr": "192.168.1.63",
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:3a:8d:f1",
            "OS-EXT-IPS:port_id": "c79d2813-94c9-4135-973e-cbf5d23e78e6",
            "OS-EXT-IPS:type": "fixed"
          }
        ]
      }
    },
    {
      "flavor": {
        "disk": "9309",
        "vcpus": "32",
        "ram": "193047",
        "id": "physical.s1.medium.ondemand",
        "name": "physical.s1.medium.ondemand"
      },
      "accessIPv4": "",
      "accessIPv6": "",
      "status": "ACTIVE",
      "progress": 0,
      "hostId": "cd243addb5d2c64e89218180b7a3ed95abe6882e81c337cc563137df",
      "updated": "2018-09-10T01:20:58Z",
      "created": "2018-09-06T09:29:27Z",
      "metadata": {

```

```
    "baremetalPortIDList": "[c79d2813-94c9-4135-973e-cbf5d23e78e6]",
    "chargingMode": "0"
  },
  "tags": [
    {
      "key": "_type_baremetal"
    }
  ],
  "description": "bms-test1",
  "locked": false,
  "config_drive": "",
  "tenant_id": "bbf1946d374b44a0a2a95533562ba954",
  "user_id": "0c50494c5816425eb05c40b5e81ab65a",
  "key_name": "$key_name",
  "OS-EXT-STS:power_state": 1,
  "OS-EXT-STS:vm_state": "active",
  "OS-EXT-SRV-ATTR:host": "bms.dc1",
  "OS-EXT-SRV-ATTR:instance_name": "instance-0014bdc2",
  "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova002@2",
  "OS-DCF:diskConfig": "MANUAL",
  "OS-EXT-AZ:availability_zone": "az-dc-1",
  "os:scheduler_hints": {},
  "OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
  "OS-EXT-SRV-ATTR:ramdisk_id": "",
  "enterprise_project_id": "0",
  "OS-SRV-USG:launched_at": "2018-09-06T09:30:36.000000",
  "OS-EXT-SRV-ATTR:kernel_id": "",
  "OS-EXT-SRV-ATTR:launch_index": 0,
  "host_status": "UP",
  "OS-EXT-SRV-ATTR:reservation_id": "r-qjad3fv0",
  "OS-EXT-SRV-ATTR:hostname": "bms-test1",
  "sys_tags": [
    {
      "key": "_sys_enterprise_project_id",
      "value": "0"
    }
  ]
},
{
  "id": "a1541cfc-8ac3-43e9-a70d-b8d4b395b256",
  "name": "bms_test2",
  "addresses": {
    "5849fdf1-9d79-4589-80c2-fe557990c417": [
      {
        "version": "4",
        "addr": "192.168.1.50",
        "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:8f:38:2c",
        "OS-EXT-IPS:port_id": "460aa585-9f83-4719-9527-fc39ebaca9aa",
        "OS-EXT-IPS:type": "fixed"
      }
    ]
  },
  "flavor": {
    "disk": "9309",
    "vcpus": "32",
    "ram": "193047",
    "id": "physical.s1.medium",
    "name": "physical.s1.medium"
  },
  "accessIPv4": "",
  "accessIPv6": "",
  "status": "SHUTOFF",
  "hostId": "cd243addb5d2c64e89218180b7a3ed95abe6882e81c337cc563137df",
  "updated": "2018-09-06T10:00:25Z",
  "created": "2018-08-30T12:40:47Z",
  "metadata": {
    "metering.order_id": "CS18083020422CNV9",
    "baremetalPortIDList": "[460aa585-9f83-4719-9527-fc39ebaca9aa]",
    "metering.product_id": "00301-167001-0--0",
    "chargingMode": "1"
  }
},
```

```
{
  "tags": [
    {
      "key": "_sys_enterprise_project_id=9dd1131d-71fd-40fe-8f14-3fe6b6b5ef8b",
      "value": "value1",
      "three=3",
      "two=2"
    }
  ],
  "description": "bms_test2",
  "locked": false,
  "config_drive": "",
  "tenant_id": "bbf1946d374b44a0a2a95533562ba954",
  "user_id": "3fc5ab2b0c544979abcaafd86edd80e6",
  "key_name": "$key_name",
  "OS-EXT-STS:power_state": 4,
  "OS-EXT-STS:vm_state": "stopped",
  "OS-EXT-SRV-ATTR:host": "bms.dc1",
  "OS-EXT-SRV-ATTR:instance_name": "instance-0014581b",
  "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova002@2",
  "OS-DCF:diskConfig": "MANUAL",
  "OS-EXT-AZ:availability_zone": "az-dc-1",
  "os:scheduler_hints": {},
  "OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
  "OS-EXT-SRV-ATTR:ramdisk_id": "",
  "enterprise_project_id": "0",
  "OS-SRV-USG:launched_at": "2018-08-30T12:42:10.000000",
  "OS-EXT-SRV-ATTR:kernel_id": "",
  "OS-EXT-SRV-ATTR:launch_index": 0,
  "host_status": "UP",
  "OS-EXT-SRV-ATTR:reservation_id": "r-i5w3yc9a",
  "OS-EXT-SRV-ATTR:hostname": "bms-test2",
  "sys_tags": [
    {
      "key": "_sys_enterprise_project_id",
      "value": "0"
    }
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.4 BMS Status Management

4.4.1 Changing a BMS Name

Function

This API is used to change the name of a BMS.

Constraints

The new name must meet the following requirements:

Contain 1 to 63 characters, consisting of letters (case-insensitive), digits, underscores (_), hyphens (-), and periods (.).

NOTE

This API can only be used to change a BMS name. The hostname will not be changed. The new name takes effect after the BMS is restarted.

URI

PUT /v1/{project_id}/baremetalservers/{server_id}

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

Table 4-36 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the BMS data structure. For details, see Table 4-37 .

Table 4-37 server data structure

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the new BMS name.

Example Request

Changing the name of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd) to **new-server-test**

```
PUT https://{BMS Endpoint}/v1/c685484a8cc2416b97260938705deb65/baremetalservers/  
95bf2490-5428-432c-ad9b-5e3406f869dd  
{  
  "server":  
  {  
    "name": "new-server-test"  
  }  
}
```

Response Parameters

Parameter	Type	Description
server	Object	Specifies the BMS information. For details, see Table 4-38 .

Table 4-38 server data structure

Parameter	Type	Description
name	String	Specifies the BMS name.
id	String	Specifies the unique ID of the BMS.
status	String	Specifies the BMS status. • ACTIVE: Running, Stopping, Deleting • BUILD: Creating • ERROR: Faulty • HARD_REBOOT: Forcibly Restarting • REBOOT: Restarting • SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen
created	String	Specifies the time when the BMS was created. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T03:30:52Z.

Parameter	Type	Description
updated	String	Specifies the last time when the BMS was updated. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T04:30:52Z.
flavor	Object	Specifies the BMS flavor. For details, see Table 4-41 .
image	Object	Specifies the BMS image. For details, see Table 4-42 .
tenant_id	String	Specifies the ID of the tenant owning the BMS. The value is in UUID format. This parameter specifies the same meaning as project_id .
key_name	String	Specifies the SSH key name.
user_id	String	Specifies the ID of the user owning the BMS.
metadata	Object	Specifies the BMS metadata. The structure is Map<String, String>. For details, see Table 4-44 .
hostId	String	Specifies the host ID of the BMS.
addresses	Object	Specifies the network address of the BMS. The structure is Map<String, Object>. For details, see Table 4-45 .
security_groups	Array of objects	Specifies security groups of the BMS. For details, see Table 4-47 .
links	Array of objects	Specifies shortcut links of the BMS. For details, see Table 4-43 .

Parameter	Type	Description
OS-DCF:diskConfig	String	Specifies the disk configuration method. This is an extended attribute. The value can be: • MANUAL: The BMS is created using the partitioning scheme and file system in the image. If the target flavor has a large disk, the remaining disk space cannot be partitioned. • AUTO: The BMS is created using a single partition with the same size as the disk of the target flavor. The file system can automatically adapt to the entire partition.
OS-EXT-AZ:availability_zone	String	Specifies the AZ where the BMS is.
OS-EXT-SRV-ATTR:host	String	Specifies the hostname of the BMS. This is an extended attribute.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the hostname of the hypervisor.
OS-EXT-SRV-ATTR:instance_name	String	Specifies the BMS alias. This is an extended attribute.
OS-EXT-STS:power_state	Integer	Specifies the power status of the BMS. This is an extended attribute. Examples: • 0: indicates NO STATE. • 1: indicates RUNNING. • 4: indicates SHUTDOWN.

Parameter	Type	Description
OS-EXT-STS:task_state	String	Specifies the task status of the BMS. This is an extended attribute. Examples: • rebooting: The BMS is being restarted. • reboot_started: The BMS is normally restarted. • reboot_started_hard: The BMS is forcibly restarted. • powering-off: The BMS is being powered off. • powering-on: The BMS is being powered on. • rebuilding: The BMS is being rebuilt. • scheduling: The BMS is being scheduled. • deleting: The BMS is being deleted.
OS-EXT-STS:vm_state	String	Specifies the BMS status. This is an extended attribute. Examples: • active: The BMS is running. • stopped: The BMS is stopped.
OS-SRV-USG:launched_at	String	Specifies the time when the BMS was started. This is an extended attribute. The timestamp format is ISO 8601, for example, 2019-05-25T03:40:25.000000 .
OS-SRV-USG:terminated_at	String	Specifies the time when the BMS was stopped. This is an extended attribute. The timestamp format is ISO 8601, for example, 2019-06-25T03:40:25.000000 .
os-extended-volumes:volumes_attached	Array of objects	Specifies EVS disks attached to the BMS. For details, see Table 4-48 .
accessIPv4	String	This is a reserved attribute.
accessIPv6	String	This is a reserved attribute.
fault	Object	Specifies the cause of a BMS fault. For details, see Table 4-49 .
config_drive	String	This is a reserved attribute.
progress	Integer	This is a reserved attribute.
description	String	Specifies description about the BMS.

Parameter	Type	Description
host_status	String	Specifies the status of the BMS host. • 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 the maintenance state. • Empty string: There is no BMS host information.
OS-EXT-SRV-ATTR:hostname	String	Specifies the host name of the BMS.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies reserved IDs for multiple BMS s created in a batch. For BMSs created in a batch, they have the same reservation_id .
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the startup sequence for BMSs created in a batch.
OS-EXT-SRV-ATTR:kernel_id	String	Specifies the UUID of the kernel image when the 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 when the AMI image is used. In other scenarios, leave this parameter blank.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the BMS system disk, for example, /dev/sdb .
OS-EXT-SRV-ATTR:user_data	String	Specifies the user_data specified during BMS creation. The value is encoded using Base64 or an empty string.
locked	Boolean	Specifies whether the BMS is locked. • true: The BMS is locked. • false: The BMS is not locked.
tags	Array of strings	Specifies tags of the BMS.
sys_tags	Array of objects	Specifies system tags of the BMS. For details, see Table 4-39 .
enterprise_project_id	String	Specifies the enterprise project ID.

Parameter	Type	Description
os:scheduler_hints	Object	Specifies the BMS scheduling information. For details, see Table 4-40 .

Table 4-39 sys_tags data structure

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

Table 4-40 os:scheduler_hints data structure

Parameter	Type	Description
group	Array of strings	Specifies the BMS group ID in UUID format.

Table 4-41 flavor data structure

Parameter	Type	Description
id	String	Specifies the flavor ID.
links	Array of objects	Specifies shortcut links of the flavor. For details, see Table 4-43 .

Table 4-42 image data structure

Parameter	Type	Description
id	String	Specifies the image ID of the BMS.
links	Array of objects	Specifies shortcut links of the BMS image. For details, see Table 4-43 .

Table 4-43 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Table 4-44 metadata data structure

Parameter	Type	Description
key	String	Specifies the key and value pair of the metadata. Each key or value contains a maximum of 255 characters.

Table 4-45 addresses data structure

Parameter	Type	Description
vpc_id	Array of objects	Specifies the VPC where the BMS is. • key: indicates the VPC ID. • value: indicates the VPC details. For details, see Table 4-46.

Table 4-46 address data structure

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	Specifies the type of the IP address. The value can be 4 or 6 . • 4: The type of the IP address is IPv4. • 6: The type of the IP address is IPv6.
OS-EXT-IPS-MAC:mac_addr	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.
OS-EXT-IPS:port_id	String	Specifies the port ID corresponding to the IP address.

Table 4-47 security_groups data structure

Parameter	Type	Description
name	String	- If no security group is specified during the BMS creation, the default security group is used. - When creating a BMS, you need to specify the ID of an existing security group (in UUID format).

Table 4-48 os-extended-volumes:volumes_attached data structure

Parameter	Type	Description
id	String	Specifies the EVS disk ID.
delete_on_termination	Boolean	Specifies whether to delete the disk when deleting the BMS. true: Yes false: No

Table 4-49 fault data structure

Parameter	Type	Description
message	String	Specifies the fault information.
code	Integer	Specifies the fault code.
details	String	Specifies the fault details.
created	String	Specifies the time when a fault occurred. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T03:30:52Z.

Example Response

```
{
  "server": {
    "tenant_id": "c685484a8cc2416b97260938705deb65",
    "addresses": {
      "08a7715f-7de6-4ff9-a343-95ba4209f24a": [
        {
          "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:0e:c4:77",
          "OS-EXT-IPS:type": "fixed",
          "OS-EXT-IPS:port_id": "0c111191-9785-49e6-8b42-66ef2802bd8f",
          "addr": "192.168.0.107",
          "version": 4
        }
      ]
    },
    "metadata": {
      "op_svc_userid": "1311c433dd9b408886f57d695c229cbe"
    },
    "OS-EXT-STS:task_state": null,
    "OS-DCF:diskConfig": "MANUAL",
    "OS-EXT-AZ:availability_zone": "az-dc-1",
    "links": [
      {
        "rel": "self",
        "href": "https://bms.az0.dc1.domainname.com/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
      },
      {
        "rel": "bookmark",
        "href": "https://bms.az0.dc1.domainname.com/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
      }
    ],
    "OS-EXT-STS:power_state": 1,
    "id": "95bf2490-5428-432c-ad9b-5e3406f869dd",
    "os-extended-volumes:volumes_attached": [
      {
        "id": "dfa375b5-9856-44ad-a937-a4802b6434c3"
      },
      {
        "id": "bb9f1b27-843b-4561-b62e-ca18eeaec417"
      },
      {
        "id": "86e801c3-acc6-465d-890c-d43ba493f553"
      },
      {
        "id": "0994d3ac-3c6a-495c-a439-c597a4f08fa6"
      }
    ],
    "OS-EXT-SRV-ATTR:host": "bms.az-dc-1",
    "image": {
      "links": [
        {
          "rel": "bookmark",
          "href": "https://bms.az0.dc1.domainname.com/c685484a8cc2416b97260938705deb65/images/1a6635d8-afea-4f2b-abb6-27a202bad319"
        }
      ],
      "id": "1a6635d8-afea-4f2b-abb6-27a202bad319"
    },
    "OS-SRV-USG:terminated_at": null,
    "accessIPv4": "",
    "accessIPv6": "",
    "created": "2017-05-24T06:14:05Z",
    "hostId": "e9c3ee0fcc58ab6085cf30df70b5544eab958858fb50d925f023e53e",
    "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova004@2",
    "key_name": "$key_name",
    "flavor": {
      "links": [
```

```
{
  "rel": "bookmark",
  "href": "https://bms.az0.dc1.domainname.com/c685484a8cc2416b97260938705deb65/flavors/physical.83.medium"
},
{
  "id": "physical.83.medium"
},
{
  "security_groups": [
    {
      "name": "0011b620-4982-42e4-ad12-47c95ca495c4"
    }
  ],
  "config_drive": "",
  "OS-EXT-STS:vm_state": "active",
  "OS-EXT-SRV-ATTR:instance_name": "instance-0000ebd3",
  "user_id": "1311c433dd9b408886f57d695c229cbe",
  "name": "bms-83",
  "progress": 0,
  "OS-SRV-USG:launched_at": "2017-05-25T03:40:25.066078",
  "updated": "2017-05-25T03:40:25Z",
  "status": "ACTIVE"
}
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.4.2 Reinstalling a BMS OS

Function

This API is used to reinstall the BMS OS. The OS of BMSs supporting quick provisioning can be reinstalled using the original image without any change to the data disks. Password and key pair injection are supported during OS reinstallation.

NOTE

To check whether a BMS is quickly provisioned, use the [Querying Flavor Details and Extended Flavor Information](#) API.

This is an asynchronous API. Calling the API successfully indicates that the task is delivered successfully. To check whether the task is successful, use the [Querying Task Statuses](#) API.

Constraints

- For BMSs created from private images, ensure that Cloud-Init (for Linux) or Cloudbase-Init (for Windows) has been installed for the image. If an image

without Cloud-Init or Cloudbase-Init is used, this API cannot inject a key pair or password. Cloud-Init or Cloudbase-Init has been installed for public images by default.

- You are not allowed to perform other operations when reinstalling the OS. Otherwise, reinstalling the OS will fail.
- You can reinstall the OS only on a BMS that is stopped or for which OS installation has failed.
- Windows images do not support password injection. An encrypted password will be used as the value of **user_data** for Linux images.

URI

POST /v1/{project_id}/baremetalservers/{server_id}/reinstall

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

Table 4-50 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
os-reinstall	Yes	Object	Specifies the operation of reinstalling the BMS OS. For details, see Table 4-51 .

Table 4-51 os-reinstall data structure

Parameter	Mandatory	Type	Description
adminpass	No	String	Specifies the initial login password of the BMS administrator account. The Linux administrator is root, and the Windows administrator is Administrator. Recommended password complexity requirements are as follows: • The password contains 8 to 26 characters. • Contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters <code>!@\$%^_-=+[{ }],./?</code> • The password cannot contain the username or the username in reverse. NOTE • For Windows BMSs, the password cannot contain more than two consecutive characters in the username. • For Linux BMSs, user_data can be used to inject a password. In this case, adminpass is invalid. • Either adminpass or keyname can be set. • If both adminpass and keyname are empty, user_data in metadata must be set.
keyname	No	String	Specifies the key pair name. You can create a key pair using the Creating or Importing an SSH Key Pair (Native OpenStack API) API, or query existing key pairs using the Querying SSH Key Pairs (Native OpenStack API) API.
userid	No	String	Specifies the user ID. You can obtain the user ID from My Credential on the management console.
metadata	No	Object	Specifies the BMS metadata. For details, see Table 4-52.

Table 4-52 metadata data structure

Parameter	Mandatory	Type	Description
user_data	No	String	Specifies the Linux image root password injected during the BMS OS reinstallation. It is a user-defined initial password. Note: The password change script must be encoded using Base64. Recommended password complexity requirements are as follows: • Contains 8 to 26 characters. • Contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^_-=+[{ }];,./? An example is as follows: <pre>#!/bin/bash echo 'root:\$6\$V6azyeLwcD3CHlpY\$BN3VVq18fmCkj66B4zdHLWevcxlig/GpOkLcOhab9smJoLKym/Tf9Hcwa6DpiPDhdHfGEAPajFmLZa0YDd910' chpasswd -e</pre> • This script must be encoded using Base64. where, \$6\$V6azyeLwcD3CHlpY\$BN3VVq18fmCkj66B4zdHLWevcxlig is the ciphertext password, which can be generated as follows: 1. Generate an encrypted salt value. <pre>[root@test linux]# python -c "import crypt, getpass, pwd;print crypt.mksalt('\$6\$V6azyeLwcD3CHlpY')</pre> 2. Generate a ciphertext password based on the salt value. <pre>[root@test linux]# python -c "import crypt, getpass, pwd;print crypt.crypt('Cloud.1234','\'\$6\$V6azyeLwcD3CHlpY')"</pre><pre>\$6\$V6azyeLwcD3CHlpY\$BN3VVq18fmCkj66B4zdHLWevcxlig/GpOkLcOhab9smJoLKym/Tf9Hcwa6DpiPDhdHfGEAPajFmLZa0YDd910</pre> 3. After the ciphertext is generated, the password change script must be encoded using Base64.

Example Request

Reinstalling the OS of a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b; key name: KeyPair-350b; user ID: 7e25b1da389f4697a79df3a0e5bd494e)

POST https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/reinstallos

```
{
  "os-reinstall": {
    "keyname": "$key_name",
    "userid": "7e25b1da389f4697a79df3a0e5bd494e",
    "metadata": {
      "user_data": "$USER_DATA"
    }
  }
}
```

Response Parameters

Table 4-53 Normal response

Parameter	Type	Description
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

Table 4-54 Abnormal response

Parameter	Type	Description
error	Dictionary data structure	Specifies the error returned when a task submission encounters an exception. For details, see Table 4-55 .

Table 4-55 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.4.3 Starting BMSs

Function

This API is used to start BMSs of specified IDs. You can start a maximum of 1000 BMSs at a time.

NOTE

This is an asynchronous API. Calling the API successfully indicates that the task is delivered successfully. To check whether the task is successful, call the [Querying Task Statuses](#) API.

URI

POST /v1/{project_id}/baremetalservers/action

[Table 4-56](#) lists the parameters.

Table 4-56 Parameter description

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

Request Parameters

NOTE

In the request, the command to start BMSs must be sent with the parameter **os-start**. For details, see the example request.

Parameter	Mandatory	Type	Description
os-start	Yes	Object	Specifies the operation to start BMSs. For details, see Table 4-57 .

Table 4-57 os-start data structure

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies the BMS IDs. For details, see Table 4-58 .

Table 4-58 servers data structure

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Example Request

Starting two BMSs (IDs: 616fb98f-46ca-475e-917e-2563e5a8cd19 and 726fb98f-46ca-475e-917e-2563e5a8cd20)

```
POST https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/action
{
  "os-start": {
    "servers": [
      {
        "id": "616fb98f-46ca-475e-917e-2563e5a8cd19"
      },
      {
        "id": "726fb98f-46ca-475e-917e-2563e5a8cd20"
      }
    ]
  }
}
```

Response Parameter

- Normal response

Table 4-59 Normal response

Parameter	Type	Description
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

- Abnormal response

Table 4-60 Abnormal response

Parameter	Type	Description
error	Dictionary data structure	Specifies the error returned when a task submission encounters an exception. For details, see Table 4-61 .

Table 4-61 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.4.4 Restarting BMSs

Function

This API is used to restart BMSs of specified IDs. You can restart a maximum of 1000 BMSs at a time.

 NOTE

This is an asynchronous API. Calling the API successfully indicates that the task is delivered successfully. To check whether the task is successful, call the [Querying Task Statuses](#) API.

URI

POST /v1/{project_id}/baremetalservers/action

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

Table 4-62 Parameter description

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

Request Parameters

 NOTE

In the request, the command to restart BMSs must be sent with the parameter **reboot**. For details, see the example request.

Parameter	Mandatory	Type	Description
reboot	Yes	Object	Specifies the operation to restart BMSs. For details, see Table 4-63 .

Table 4-63 reboot data structure

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the BMS reboot type. • SOFT: soft restart (invalid) • HARD: hard restart (default)
servers	Yes	Array of objects	Specifies BMS IDs. For details, see Table 4-64 .

Table 4-64 servers data structure

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Example Request

Restarting two BMSs (IDs: are 616fb98f-46ca-475e-917e-2563e5a8cd19 and 726fb98f-46ca-475e-917e-2563e5a8cd20)

POST https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/action

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

Response Parameters

- Normal response

Table 4-65 Normal response

Parameter	Type	Description
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

- Abnormal response

Table 4-66 Abnormal response

Parameter	Type	Description
error	Dictionary data structure	Specifies the error returned when a task submission encounters an exception. For details, see Table 4-67 .

Table 4-67 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.4.5 Stopping BMSs

Function

This API is used to stop BMSs of specified IDs. You can stop a maximum of 1000 BMSs at a time.

NOTE

This is an asynchronous API. Calling the API successfully indicates that the task is delivered successfully. To check whether the task is successful, use the [Querying Task Statuses](#) API.

URI

POST /v1/{project_id}/baremetalservers/action

[Table 4-68](#) lists the parameters.

Table 4-68 Parameter description

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

Request Parameters

NOTE

In the request, the command to stop BMSs must be sent with the parameter **os-stop**. For details, see the example request.

Parameter	Mandatory	Type	Description
os-stop	Yes	Object	Specifies the operation to stop BMSs. For details, see Table 4-69 .

Table 4-69 os-stop data structure

Parameter	Mandatory	Type	Description
servers	Yes	Array of objects	Specifies BMS IDs. For details, see Table 4-70 .

Parameter	Mandatory	Type	Description
type	No	String	Specifies how BMSs are stopped. The default value is HARD . Value: HARD : forcible stop

Table 4-70 servers data structure

Parameter	Mandatory	Type	Description
id	Yes	String	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Example Request

Stopping two BMSs (IDs: are 616fb98f-46ca-475e-917e-2563e5a8cd19 and 726fb98f-46ca-475e-917e-2563e5a8cd20)

POST https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/action

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

Response Parameters

- Normal response

Table 4-71 Normal response

Parameter	Type	Description
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

- Abnormal response

Table 4-72 Abnormal response

Parameter	Type	Description
error	Dictionary data structure	Specifies the error returned when a task submission encounters an exception. For details, see Table 4-73 .

Table 4-73 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.5 BMS Metadata Management

4.5.1 Updating BMS Metadata

Function

This API is used to update BMS metadata.

- If the metadata does not contain the target field, the field is automatically added to the field.
- 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

The BMS must be in active, stopped, or paused state. The state is indicated by the **OS-EXT-STS:vm_state** parameter.

URI

POST /v1/{project_id}/baremetalservers/{server_id}/metadata

[Table 4-74](#) lists the parameters.

Table 4-74 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Table 4-75 Request parameters

Parameter	Mandatory	Type	Description
metadata	Yes	Object	Specifies the user-defined metadata key-value pair. If you do not specify any key-value pair, metadata will not be updated. key: It contains a maximum of 255 Unicode characters which can be letters, digits, hyphens (-), underscores (_), colons (:), and periods (.). key cannot be left blank. value: It contains a maximum of 255 Unicode characters.

Example Request

Updating the metadata of a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b) to customize a key-value pair

```
POST https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/metadata
{
  "metadata": {
    "key": "value"
  }
}
```

Response Parameters

Table 4-76 Parameter description

Parameter	Mandatory	Type	Description
metadata	Yes	Object	Specifies the user-defined metadata key-value pair. If you do not specify any key-value pair, metadata will not be updated. key: It contains a maximum of 255 Unicode characters which can be letters, digits, hyphens (-), underscores (_), colons (:), and periods (.). key cannot be left blank. value: It contains a maximum of 255 Unicode characters.

Example Response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.6 BMS NIC Management

4.6.1 Querying NICs Bound to a BMS

Function

This API is used to query information about NICs bound to a BMS, such as the IP address and MAC address of each NIC.

URI

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

[Table 4-77](#) lists the parameters.

Table 4-77 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying information about NICs bound to a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)

```
GET https://{BMS Endpoint}/v1/c685484a8cc2416b97260938705deb65/baremetalservers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-interface
```

Response Parameters

Parameter	Type	Description
interfaceAttachments	Array of objects	Specifies BMS NICs. For details, see Table 4-78 .

Table 4-78 interfaceAttachments data structure

Parameter	Type	Description
port_state	String	Specifies the NIC port status. The value can be ACTIVE , BUILD , or DOWN .
fixed_ips	Array of objects	Specifies private IP addresses of NICs. For details, see Table 4-79 .
net_id	String	Specifies the ID of the subnet (network_id) to which the NIC ports belong.
port_id	String	Specifies the ID of the NIC port.
mac_addr	String	Specifies the MAC address of the NIC.
driver_mode	String	Specifies the NIC driver type in Guest OS. The value can be virtio or hinic . The default value is virtio .
pci_address	String	Specifies the BDF number of the NIC in Linux Guest OS.

Table 4-79 fixed_ips data structure

Parameter	Type	Description
subnet_id	String	Specifies the ID of the subnet (subnet_id) corresponding to the private IP address of the NIC.
ip_address	String	Specifies the NIC private IP address.

Example Response

```
{
  "interfaceAttachments": [
    {
      "port_state": "ACTIVE",
      "fixed_ips": [
        {
          "subnet_id": "a5052101-11e1-4f3d-a5fa-f6ba6791219c",
          "ip_address": "192.168.1.147"
        }
      ],
      "net_id": "ad0fadbf-4bc1-472c-a030-5310e53b3818",
      "port_id": "bb585b04-f2a2-4528-9064-fd0aeb4e15a9",
      "mac_addr": "fa:16:3e:3b:58:fc"
    }
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.7 BMS Disk Management

4.7.1 Attaching an EVS Disk to a BMS

Function

This API is used to attach EVS disks to a BMS as data disks after the BMS is created if existing data disks are insufficient or cannot meet requirements.

Constraints

- EVS disks cannot be attached to a BMS in a batch.
- A maximum of 60 EVS disks can be attached to a BMS.
- A bootable disk cannot be attached to a BMS.
- A disk cannot be attached to a BMS in the **PAUSED** state, which is specified using the **OS-EXT-STS:vm_state** parameter.
- Only a shared disk or a disk in the **available** state can be attached to a BMS.
- Only EVS disks whose device type is **SCSI** can be attached to a BMS.

URI

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

[Table 4-80](#) lists the parameters.

Table 4-80 Parameter description

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

Parameter	Mandatory	Description
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
volumeAttachment	Yes	Object	Specifies the disks to be attached. For details, see Table 4-81 .

Table 4-81 volumeAttachment data structure

Parameter	Mandatory	Type	Description
volumeId	Yes	String	Specifies the ID of the disk to be attached to a BMS. You can obtain the disk ID from the EVS console or by calling the Querying EVS Disks .
device	No	String	Specifies the mount point, such as /dev/sda and /dev/sdb . The new disk mount point cannot be the same as an existing one. The mount point must be specified based on the sequence of existing device names. If this parameter is left blank or set to "", the system automatically generates a mount point.

Example Request

Attaching an EVS disk (ID: b53f23bd-ee8f-49ec-9420-d1acfeaf91d6) to a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b)

```
POST https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/attachvolume
{
  "volumeAttachment": {
    "volumeId": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "device": ""
  }
}
```

Response Parameters

Table 4-82 Normal response

Parameter	Type	Description
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

Table 4-83 Abnormal response

Parameter	Type	Description
error	Dictionary data structure	Specifies the error returned when a task submission encounters an exception. For details, see Table 4-84 .

Table 4-84 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

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

Returned Values

Normal values

Returned Value	Description
202	The request has been accepted, but the processing has been delayed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.7.2 Detaching an EVS Disk from a BMS

Function

This API is used to detach a disk from a BMS.

- A disk attached to **/dev/sda** functions as the system disk. You can only detach the system disk from a stopped BMS.
- Disks attached to a mount point other than **/dev/sda** function as data disks and can be detached from a running or stopped BMS.

Constraints

If a BMS is stopped, disks can be detached from it without any constraint on the OS. If a BMS is running, the constraints are as follows:

- Before detaching an EVS disk from a Linux BMS, log in to the BMS, run **umount** to unmount the disk from the file system, and ensure that no program is reading data from or writing data to the disk. Otherwise, the disk will fail to be detached.
- Before detaching an EVS disk from a Windows BMS, ensure that no program is reading data from or writing data to the disk. Otherwise, data will be lost.

URI

DELETE /v1/{project_id}/baremetalservers/{server_id}/detachvolume/{attachment_id}

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

Table 4-85 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Parameter	Mandatory	Description
attachment_id	Yes	Specifies the IDs of EVS disks attached to the BMS. You can obtain the IDs of EVS disks attached to a BMS using the Querying EVS Disks Attached to a BMS API.

Request Parameters

None

Example Request

Detaching an EVS disk (ID: 6b604cef-9bd8-4f5a-ae56-45839e6e1f0a) from a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b)
DELETE https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/detachvolume/6b604cef-9bd8-4f5a-ae56-45839e6e1f0a

Response

See [Task ID Response](#).

Returned Values

Normal values

Returned Value	Description
202	The request has been accepted, but the processing has been delayed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.7.3 Querying EVS Disks Attached to a BMS

Function

This API is used to query EVS disks attached to a BMS.

URI

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

[Table 4-86](#) lists the parameters.

Table 4-86 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying EVS disks attached to a BMS (ID: 4d8c3732-a248-40ed-bebc-539a6ffd25c0)

```
GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/4d8c3732-a248-40ed-bebc-539a6ffd25c0/os-volume_attachments
```

Response Parameters

Parameter	Type	Description
volumeAttachments	Array of objects	Specifies disks attached to a BMS. For details, see Table 4-87 .

Table 4-87 volumeAttachments data structure

Parameter	Type	Description
device	String	Specifies the mount directory, for example, dev/sdd .
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ID of the BMS that disks are attached to.
volumeId	String	Specifies the ID of the disk attached to the BMS.

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

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.8 BMS Password Management

4.8.1 Querying Whether One-Click Password Reset Is Supported

Function

This API is used to query whether a BMS supports one-click password reset.

URI

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

[Table 4-88](#) lists the parameters.

Table 4-88 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying whether a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b) supports one-click password reset

```
GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/os-resetpwd-flag
```

Response Parameters

Parameter	Type	Description
resetpwd_flag	String	Specifies whether one-click password reset is supported. • True: One-click password reset is supported. • False: One-click password reset is not supported.

Example Response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.8.2 Obtaining the Password of a Windows BMS

Function

This API is used to query the random password of the administrator (account **Administrator** or an account configured using Cloudbase-Init) generated by the system during the installation of a Windows BMS.

If the BMS is created from a private image, ensure that Cloudbase-Init has been installed. Cloudbase-Init is installed on public images by default.

URI

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

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

Table 4-89 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Obtaining the password of a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b)
GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/os-server-password

Response Parameters

Parameter	Type	Description
password	String	Specifies the password in ciphertext.

Example Response

```
{
  "password": "UHC9+YW1xDC1Yu8Mg9n+tnOp7euEO/cW//9KgJJKWhr5w=="
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.8.3 Deleting the Password of a Windows BMS

Function

This API is used to delete the random password generated during initial Windows BMS installation. After the password is deleted, you can still use your password to log in to your BMS. However, you cannot use the Get Password function to recover the BMS initial password.

If the BMS is created from a private image, ensure that Cloudbase-Init has been installed. Cloudbase-Init is installed on public images by default.

URI

DELETE /v1/{project_id}/baremetalservers/{server_id}/os-server-password

[Table 4-90](#) lists the parameters.

Table 4-90 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Deleting the password of a BMS (ID: cf2a8b97-b5c6-47ef-9714-eb27adf26e5b)
DELETE https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/cf2a8b97-b5c6-47ef-9714-eb27adf26e5b/os-server-password

Response

N/A

Returned Values

Normal values

Returned Value	Description
202	The request has been accepted, but the processing has been delayed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.9 BMS Quota Management

4.9.1 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}/baremetalservers/limits

[Table 4-91](#) lists the parameters.

Table 4-91 Parameter description

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

Request Parameters

None

Example Request

Querying the quotas of a tenant (project ID:
bbf1946d374b44a0a2a95533562ba954)

GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/baremetalservers/limits

Response Parameters

Parameter	Type	Description
absolute	Object	Specifies tenant quotas. For details, see Table 4-92 .

NOTE

Value -1 indicates that the quantity is unlimited.

Table 4-92 absolute data structure

Parameter	Type	Description
maxTotalInstances	Integer	Specifies the maximum number of BMSs you can create.

Parameter	Type	Description
maxTotalCores	Integer	Specifies the maximum number of CPU cores you can use.
maxTotalRAMSize	Integer	Specifies the maximum memory (MB) you can use.
maxTotalKeypairs	Integer	Specifies the maximum number of SSH key pairs you can use.
maxServerMeta	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 BMSs in a server 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 limit 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 limit complies with the VPC quota limit.
maxTotalFloatingIps	Integer	Specifies the maximum number of EIPs you can use.
maxImageMeta	Integer	Specifies the maximum length of the image metadata.
totalInstancesUsed	Integer	Specifies the number of the used BMSs.
totalCoresUsed	Integer	Specifies the number of used CPU cores.
totalRAMUsed	Integer	Specifies the used memory (MB).
totalSecurityGroupsUsed	Integer	Specifies the number of used security groups.
totalFloatingIpsUsed	Integer	Specifies the number of used EIPs.

Example Response

```
{
  "absolute": {
    "maxServerMeta": 128,
    "maxPersonality": 5,
    "maxImageMeta": 128,
    "maxPersonalitySize": 10240,
    "maxSecurityGroupRules": 20,
    "maxTotalKeypairs": 100,
    "totalRAMUsed": 799836,
    "totalInstancesUsed": 21,
    "maxSecurityGroups": 10,
    "totalFloatingIpsUsed": 0,
    "maxTotalCores": -1,
    "totalSecurityGroupsUsed": 1,
    "maxTotalFloatingIps": 10,
    "maxTotalInstances": 100,
    "totalCoresUsed": 148,
    "maxTotalRAMSize": -1,
    "maxServerGroups": -1,
    "maxServerGroupMembers": -1,
    "totalServerGroupsUsed": 1
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.10 BMS Flavor Management

4.10.1 Querying Flavor Details and Extended Flavor Information

Function

This API is used to query BMS flavor details and extended flavor information. You can call this API to query the value of parameter **baremetal:extBootType** to check whether a flavor supports quick BMS provisioning.

URI

GET /v1/{project_id}/baremetalservers/flavors

[Table 4-93](#) lists the parameters.

Table 4-93 Parameter description

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

Request Parameters

None

Example Request

Querying BMS flavor details and extended flavor information in the **az1** AZ
GET https://{BMS Endpoint}/v1/c685484a8cc2416b97260938705deb65/baremetalservers/flavors?
availability_zone=az1

Response Parameters

Parameter	Type	Description
flavors	Array of objects	Specifies BMS flavors. For details, see Table 4-94 .

Table 4-94 flavors data structure

Parameter	Type	Description
id	String	Specifies the ID of a BMS flavor.
name	String	Specifies the name of a BMS flavor.
vcpus	String	Specifies the number of CPU cores in a BMS flavor.
ram	Integer	Specifies the memory size (MB) in a BMS flavor.
disk	String	Specifies the system disk size in a BMS flavor. The value 0 indicates that the disk size is not limited.
swap	String	This is a reserved attribute.

Parameter	Type	Description
OS-FLV-EXT-DATA:ephemeral	Integer	This is a reserved attribute.
OS-FLV-DISABLED:disabled	Boolean	This is a reserved attribute.
rxtx_factor	Float	This is a reserved attribute.
rxtx_quota	String	This is a reserved attribute.
rxtx_cap	String	This is a reserved attribute.
os-flavor-access:is_public	Boolean	Specifies whether a flavor is public. false indicates a private flavor and true indicates a public flavor.
links	Array of objects	Specifies shortcut links of a flavor. For details, see Table 4-95 .
os_extra_specs	Object	Specifies extended fields of a BMS flavor. For details, see Table 4-96 .

Table 4-95 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.
type	String	Specifies the shortcut link type.

Table 4-96 os_extra_specs data structure

Parameter	Type	Description
resource_type	String	Specifies the resource type of a flavor.

Parameter	Type	Description
capabilities:cpu_arch	String	Specifies the CPU architecture of the BMS. The value can be: • x86_64 (applicable to x86 servers) • aarch64 (applicable to Arm servers)
baremetal:disk_detail	String	Specifies physical disk specifications.
capabilities:hypervisor_type	String	Specifies a flavor of the Ironic type.
baremetal:__support_evs	String	Specifies whether a flavor supports EVS disks. • true • false If the flavor of a BMS does not contain this parameter, EVS disks cannot be attached to the BMS.
baremetal:extBootType	String	Specifies the boot source of the BMS. • LocalDisk: local disk • Volume: EVS disk (quick provisioning)
capabilities:board_type	String	Specifies the type of a BMS flavor in the format of flavor abbreviation. For example, if the flavor name is physical.o2.medium , the flavor type is o2m .
baremetal:net_num	String	Specifies the maximum number of NICs on the BMS.
baremetal:netcard_detail	String	Specifies physical NIC specifications.
baremetal:cpu_detail	String	Specifies physical CPU specifications.
baremetal:memory_detail	String	Specifies physical memory specifications.

Parameter	Type	Description
cond:operation:status	String	Specifies the status of a BMS flavor. If this parameter is not set, its default value is normal. • normal: indicates normal commercial use of the flavor. • abandon: indicates that the flavor has been disabled (not displayed). • sellout: indicates that the flavor has been sold out. • obt: indicates that the flavor is under OBT. • promotion: indicates the recommended flavor (commercial use, which is similar to normal).
cond:operation:az	String	Specifies the status of a BMS flavor in an AZ. 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. Its format is az (xx). xx indicates the status of a BMS flavor in an AZ, and it is mandatory. For example, a flavor is for commercial use in AZ0 and AZ3, sold out in AZ1, for OBT in AZ2, and is canceled in other AZs. Then, set parameters as follows: • cond:operation:status: abandon • cond:operation:az: az0(normal), az1(sellout), az2(obt), az3(promotion) NOTE Configure this parameter if the flavor status in an AZ is different from the cond:operation:status value.

Example Response

```
{
  "flavors": [
    {
      "id": "physical.kl1.3xlarge",
      "name": "physical.kl1.3xlarge",
      "vcpus": "24",
      "ram": 321729,
      "disk": "6707",
```

```
    "swap": "",
    "links": [
      {
        "rel": "self",
        "href": "https://compute.Region.dc1.domainname.com/v2/
bbf1946d374b44a0a2a95533562ba954/flavors/physical.kl1.3xlarge",
        "type": null
      },
      {
        "rel": "bookmark",
        "href": "https://compute.Region.dc1.domainname.com/bbf1946d374b44a0a2a95533562ba954/
flavors/physical.kl1.3xlarge",
        "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": false,
    "os_extra_specs": {
      "capabilities:cpu_arch": "x86_64",
      "baremetal:disk_detail": "SAS SSD:2*800G Raid 1 + NVMe SSD Card1.6T",
      "capabilities:hypervisor_type": "ironic",
      "baremetal:__support_evs": "true",
      "baremetal:extBootType": "LocalDisk",
      "capabilities:board_type": "o2m",
      "baremetal:net_num": "2",
      "baremetal:netcard_detail": "2 x 2*10GE",
      "baremetal:cpu_detail": "Intel Xeon E5-2667 V4 (2*8core* 3.2 GHz)",
      "resource_type": "ironic",
      "baremetal:memory_detail": "256GB DDR4 RAM(GB)"
    }
  }
]
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.11 Task Management

4.11.1 Querying Task Statuses

Function

This API is used to query the execution status of a task.

After an asynchronous API is called, for example an API for creating BMSs or attaching or detaching disks, **job_id** will be returned. You can use **job_id** to query the task status.

URI

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

[Table 4-97](#) lists the parameters.

Table 4-97 Parameter description

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

Request Parameters

None

Example Request

Querying the status of a task (ID: 2c9eb2c5544cbf6101544f0635672b60)
GET https://{BMS Endpoint}/v1/bbf1946d374b44a0a2a95533562ba954/jobs/
2c9eb2c5544cbf6101544f0635672b60

Response Parameters

Parameter	Type	Description
status	String	Specifies the task status. • SUCCESS: The task is successful. • RUNNING: The task is running. • FAIL: The task failed. • INIT: The task is being initialized.
entities	Object	Specifies the task object. For details, see Table 4-98 . The displayed information varies depending on what tasks are executed. For tasks related to BMSs, server_id is displayed. For tasks related to NICs, nic_id is displayed. If subtasks exist, details about the subtasks are also displayed.

Parameter	Type	Description
job_id	String	Specifies the task ID.
job_type	String	Specifies the task. The value can be: • baremetalBatchCreate: creating BMSs in a batch • baremetalBatchOperate: changing the power status of BMSs in a batch. • baremetalVolumeBootReinstallOs: reinstalling the OS of a BMS which is provisioned quickly • baremetalReinstallOs: reinstalling the OS of a BMS with local disks • baremetalAttachVolume: attaching a disk • baremetalDetachVolume: detaching a disk
begin_time	String	Specifies the time when the job started. The timestamp format is ISO 8601, for example, 2019-04-25T20:04:47.591Z .
end_time	String	Specifies the time when the job ended. The timestamp format is ISO 8601, for example, 2019-04-26T20:04:47.591Z .
error_code	String	Specifies the error code returned upon a task execution failure.
fail_reason	String	Specifies the cause of a task execution failure.
message	String	Specifies the message returned when an error occurs.
code	String	Specifies the error code returned when an error occurs. For details about error codes, see section Status Codes .

Table 4-98 entities data structure

Parameter	Type	Description
sub_jobs_total	Integer	Specifies the number of subtasks. If the task has no subtasks, the value of this parameter is 0 .

Parameter	Type	Description
sub_jobs	Array of objects	Specifies the execution information about a subtask. If the task has no subtasks, the value of this parameter is left blank. For details, see Table 4-99 .

Table 4-99 sub_jobs data structure

Parameter	Type	Description
status	String	Specifies the task status. • SUCCESS: The task is successful. • RUNNING: The task is running. • FAIL: The task failed. • INIT: The task is being initialized.
entities	Object	Specifies the task object. The displayed information varies depending on what tasks are executed. For tasks related to BMSs, server_id is displayed. For tasks related to NICs, nic_id is displayed. For details, see Table 4-100 .
job_id	String	Specifies the task ID.
job_type	String	Specifies the task type. The value can be: • baremetalSingleCreate: creating a single BMS • baremetalSingleOperate: changing the power status of a single BMS
begin_time	String	Specifies the time when the job started. The timestamp format is ISO 8601, for example, 2019-04-25T20:04:47.591Z .
end_time	String	Specifies the time when the job ended. The timestamp format is ISO 8601, for example, 2019-04-26T20:04:47.591Z .
error_code	String	Specifies the error code returned upon a task execution failure.
fail_reason	String	Specifies the cause of a task execution failure.
message	String	Specifies the message returned when an error occurs.

Parameter	Type	Description
code	String	Specifies the error code returned when an error occurs. For details about error codes, see section Status Codes .

Table 4-100 entities data structure

Parameter	Type	Description
server_id	String	If the task is a BMS-related operation, server_id is displayed.
nic_id	String	If the task is a NIC-related operation, nic_id is displayed.

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": "baremetalSingleCreate",
        "begin_time": "2019-04-25T20:04:47.591Z",
        "end_time": "2019-04-25T20:08:21.328Z",
        "error_code": null,
        "fail_reason": null
      }
    ]
  },
  "job_id": "2c9eb2c5544cbf6101544f0602af2b4f",
  "job_type": "baremetalBatchCreate",
  "begin_time": "2019-04-25T20:04:34.604Z",
  "end_time": "2019-04-25T20:08:41.593Z",
  "error_code": null,
  "fail_reason": null
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.12 BMS SSH Key Management (Native OpenStack APIs)

4.12.1 Querying SSH Key Pairs (Native OpenStack API)

Function

This API is used to query SSH key pairs.

Constraints

Pagination query is not supported.

URI

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

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

Table 4-101 Parameter description

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

Request Parameters

None

Example Request

Querying SSH key pairs in a project (ID: bbf1946d374b44a0a2a95533562ba954)
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/os-keypairs

Response Parameters

Parameter	Type	Description
keypairs	Array of objects	Specifies key pairs. For details, see Table 4-102 .

Table 4-102 keypairs data structure

Parameter	Type	Description
keypair	Object	Specifies details about a key pair. For details, see Table 4-103 .

Table 4-103 keypair data structure

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 in the key pair.

Example Response

```
{
  "keypairs": [
    {
      "keypair": {
        "fingerprint": "15:b0:f8:b3:f9:48:63:71:cf:7b:5b:38:6d:44:2d:4a",
        "name": "keypair-test",
        "type": "ssh",
        "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC+Eo/
RZRngaGtKfS7I62ZjslLO79KklKbMXi8F+KITD4bVQHhN+kV
+4gRgkgCRbdoDqoGfpaDFs877DYX9n4z6FrAlZ4PES8TNKhatifpn9NdQYWA+IkU8CuvlEKGuFpKRi/k7JLos/
gHi2hy7QUwgtRvcefvD/vgQZOVw/mGR9Q== Generated-by-Nova"
      }
    }
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.12.2 Querying an SSH Key Pair (Native OpenStack API)

Function

This API is used to query a specified SSH key pair based on the key pair name.

URI

GET /v2.1/{project_id}/os-keypairs/{keypair_name}

[Table 4-104](#) lists the parameters.

Table 4-104 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
keypair_name	Yes	Specifies the key pair name. You can obtain the key pair name by calling the Querying SSH Key Pairs (Native OpenStack API) API.

Request Parameters

None

Example Request

Querying details about a key pair (name: keypair-test)

GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/os-keypairs/keypair-test

Response Parameters

Parameter	Type	Description
keypair	Object	Specifies the SSH key pair. For details, see Table 4-105 .

Table 4-105 keypair data structure

Parameter	Type	Description
public_key	String	Specifies information about the public key in the key pair.
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. The timestamp format is ISO 8601, for example, 2019-05-07T12:06:13.681238 .
deleted	Boolean	Specifies the deleted key pair. • 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. The timestamp format is ISO 8601, for example, 2019-05-07T12:06:13.681238 .
id	String	Specifies the key pair ID.
updated_at	String	Specifies the time when the key pair was updated. The timestamp format is ISO 8601, for example, 2019-05-07T12:06:13.681238 .
user_id	String	Specifies information about the user owning the key pair.

Example Response

```
{
  "keypair": {
    "created_at": "2019-05-07T12:06:13.681238",
    "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/ye+/
kurqcPJFKKp60mToXIMyzioFxW88fJtwEWawHKAclbHWpR1t4fQ4DS+/slbX/Yd9btLVQ2tpQjodGDbM9Tr9/+/
3i6rcR+EoLqmbgCgAiGiVV6VbM2Zx79yUwd+GnQejHX8BlyZoOjCnt3NREslTcmWE9FVFy6TnLmahs3FkEO/
QGgWGkaohAJlsgaVvSWGgDn2AujKYwyDokK3dXyeX3m2Vmc3ejjqPa/C4nRrCOLko5nSgV/
9IXRx1ERlmsqZnE9usB Generated-by-Nova",
```

```
"updated_at": null,  
"user_id": "fake"  
}  
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.12.3 Creating or Importing an SSH Key Pair (Native OpenStack API)

Function

This API is used to create an SSH key pair or import a public key to generate a key pair.

After an SSH key pair is generated, download the private key to a local directory. Then, you can use this private key to log in to the BMS. For BMS security purposes, a private key can be downloaded only once. Keep it secure.

URI

POST /v2.1/{project_id}/os-keypairs

[Table 4-106](#) lists the parameters.

Table 4-106 Parameter description

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

Request Parameters

 NOTE

When creating an SSH key pair, you only need to configure **name**. When importing a public SSH key, you must also configure **public_key**.

Parameter	Mandatory	Type	Description
keypair	Yes	Object	Specifies the created or imported SSH key pair. For details, see Table 4-107 .

Table 4-107 keypair data structure

Parameter	Mandatory	Type	Description
public_key	No	String	Specifies the imported public key. The maximum size of the imported public key is 1024 bytes. Note: If the length of the public key to be imported exceeds 1024 bytes, the public key import to the BMS will fail.
name	Yes	String	Specifies the key pair name. The new key pair name cannot be the same as an existing one.

Example Request

Creating or importing an SSH key pair (name: keypair-7d7c3650-dabe-4eb0-b904-5c464453c043) with the public key **ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQC9mC3WZN9UGLxgPBpP7H5jZMc6pKwOoSgre8yun6REFktn/Kz7DUt9jaR1UJyRzHxITfCfAlgSxPdGqB/oF1suMyWgu5i0625vavLB5z5kC8Hq3qZJ9zJO1poE1kyD+htiTPWJ88e12xuH2XB/CZN9OpEiF98hAagiOE0EnOS5Q==** Generated by Nova\n

```
POST https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/os-keypairs
{
  "keypair": {
    "name": "keypair-7d7c3650-dabe-4eb0-b904-5c464453c043",
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQgQC9mC3WZN9UGLxgPBpP7H5jZMc6pKwOoSgre8yun6REFktn/
Kz7DUt9jaR1UJyRzHxITfCfAlgSxPdGqB/oF1suMyWgu5i0625vavLB5z5kC8Hq3qZJ9zJO1poE1kyD
+htiTPWJ88e12xuH2XB/CZN9OpEiF98hAagiOE0EnOS5Q== Generated by Nova\n"
  }
}
```

Response Parameters

Parameter	Type	Description
keypair	Object	Specifies the SSH key pair. For details, see Table 4-108 .

Table 4-108 keypair data structure

Parameter	Type	Description
fingerprint	String	Specifies fingerprint information about the key pair.
name	String	Specifies the key pair name.
public_key	String	Specifies the public key.
private_key	String	Specifies the private key. - 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 owning the key pair.

Example Response

```
{
  "keypair": {
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQgQC9mC3WZN9UGLxgPBpP7H5jZMc6pKwOoSgre8yun6REFktn/
Kz7DUt9jaR1UJyRzHxITfCfAlgSxPdGqB/of1suMyWgu5i0625vavLB5z5kC8Hq3qZJ9zJO1poE1kyD
+htiTtPWJ88e12xuH2XB/CZN9OpEiF98hAagiOE0EnOS5Q== Generated by Nova\n",
    "user_id": "f882feb345064e7d9392440a0f397c25",
    "name": "keypair-7d7c3650-dabe-4eb0-b904-5c464453c043",
    "fingerprint": "35:9d:d0:c3:4a:80:d3:d8:86:f1:ca:f7:df:c4:f9:d8"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

4.12.4 Deleting an SSH Key Pair (Native OpenStack API)

Function

This API is used to delete a specified SSH key pair based on the key pair name.

URI

DELETE /v2.1/{project_id}/os-keypairs/{keypair_name}

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

Table 4-109 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
keypair_name	Yes	Specifies the key pair name. You can obtain the key pair name by calling the Querying SSH Key Pairs (Native OpenStack API) API.

Request Parameters

None

Example Request

Deleting an SSH key pair (name: KeyPair-123)
DELETE https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/os-keypairs/keypair-test

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

5 Examples

5.1 Example 1: Creating a BMS

Scenarios

BMSs are recommended for key applications and services that require high performance (such as Big Data clusters and enterprise middleware systems) and a secure and reliable running environment.

This section describes how to create a BMS using the [Creating BMSs](#) API. For details about how to call the API, see [Calling APIs](#).

NOTE

Currently, only yearly/monthly BMSs can be created.

Prerequisites

- A BMS can boot from a local disk or an EVS disk. Quickly provisioned BMSs boot from an EVS disk. You can use either of the following methods to check the boot device:
 - After selecting a flavor on the [BMS console](#), if **System Disk** is displayed in the **Disk** option, a BMS of this flavor boots from an EVS disk. Otherwise, the BMS boots from a local disk.
 - Call the [Querying Flavor Details and Extended Flavor Information](#) API and check the value of **baremetal:extBootType** in the response. If the value is **LocalDisk**, the BMS is booted from a local disk. If the value is **Volume**, the BMS is booted from an EVS disk.

Create a BMS with Local Disks

The following example uses the most basic configuration required for creating a BMS with local disks.

```
{
  "server": {
    "availability_zone": "eu-west-101",
    "name": "bms-test01",
```

```
{
  "imageRef": "766a545a-02e1-433e-b1d1-733b5dc95e94",
  "flavorRef": "physical.s3.large",
  "vpcid": "8df83bf2-fd2e-4ee0-9692-c0b7736513fb",
  "nics": [
    {
      "subnet_id": "c6bb8788-4fd1-4a8f-adab-7eba3bed8616"
    }
  ],
  "extendparam": {
    "chargingMode": "prePaid",
    "periodType": "month",
    "periodNum": 1,
    "isAutoRenew": "true",
    "isAutoPay": "true",
    "regionID": "eu-west-101"
  },
  "metadata": {
    "op_svc_userid": "59781460e9e54886a7d03df7d3f3fc81"
  },
  "key_name": "$key_name"
}
```

- **availability_zone**: AZ where you want to create the BMS.
- **name**: name of the BMS, for example, **bms-test01**
- **imageRef**: ID of the image (OS) used to create the BMS. You can query image IDs on the IMS console.
- **flavorRef**: BMS flavor ID in `physical.x.x` format. You can obtain flavor IDs from the BMS console and determine the boot device of a flavor based on [Prerequisites](#). In this example, a flavor that boots from a local disk is used, for example, **physical.s3.large**.
- **vpcid**: ID of the VPC where the BMS is. You can query VPC IDs on the VPC console or using the [Querying VPCs](#) API.
- **nics.subnet_id**: ID of the VPC subnet. You can query subnet IDs on the VPC console or using the [Querying Subnets](#) API.
- **extendparam**: **chargingMode** specifies the billing mode, and **prePaid** indicates prepayment, that is, the yearly/monthly mode. **periodType** specifies the period type, and **month** indicates the monthly mode. **periodNum** specifies the number of periods, and **1** in this example indicates one month. **isAutoRenew** specifies whether to enable auto-renewal. **isAutoPay** specifies whether to automatically deduct the charges from your account balance without requiring manual payment. **regionID** specifies the ID of the region where the BMS is.
- **metadata.op_svc_userid**: user ID, which you can obtain from **My Credentials** on the management console.
- **key_name**: key pair name. To log in to the BMS using an SSH key pair, set the value to the name of an existing key pair. You can obtain key pair names from the **Key Pair** page on the Cloud Server Console.

Create a BMS Supporting Quick Provisioning

The following example uses the most basic configuration required for creating a BMS supporting quick provisioning.

```
{
  "server": {
    "availability_zone": "eu-west-101",
    "name": "bms-test02",
```

```
{
  "imageRef": "766a545a-02e1-433e-b1d1-733b5dc95e94",
  "flavorRef": "physical.h2.large",
  "root_volume": {
    "volumetype": "SAS",
    "size": 150
  },
  "vpcid": "8df83bf2-fd2e-4ee0-9692-c0b7736513fb",
  "nics": [
    {
      "subnet_id": "c6bb8788-4fd1-4a8f-adab-7eba3bed8616"
    }
  ],
  "extendparam": {
    "chargingMode": "prePaid",
    "periodType": "month",
    "periodNum": 1,
    "isAutoRenew": "true",
    "isAutoPay": "true",
    "regionID": "eu-west-101"
  },
  "metadata": {
    "op_svc_userid": "59781460e9e54886a7d03df7d3f3fc81"
  },
  "key_name": "$key_name"
}
```

Compared with creating a BMS with local disks, you need to select a flavor whose boot device is EVS disk and set system disk parameters.

- **root_volume.volumetype**: type of the system disk. **SAS** indicates high I/O. You can specify other types. For details, see the parameter descriptions in [Creating BMSs](#).
- **root_volume.size**: system disk size (GB). The value ranges from **40** to **1024**. The system disk size cannot be less than the minimum size of the system disk (**min_disk**) in the image. You can obtain the value of **min_disk** using the [Querying Images](#) API.

Create a BMS with Data Disks

You can configure data disks for BMSs by adding the **data_volumes** field to the request body. An example is as follows:

```
{
  "server": {
    "availability_zone": "eu-west-101",
    "name": "bms-test03",
    "imageRef": "766a545a-02e1-433e-b1d1-733b5dc95e94",
    "flavorRef": "physical.s3.large",
    "data_volumes": [
      {
        "volumetype": "SAS",
        "size": 10,
        "shareable": false
      }
    ],
    "vpcid": "8df83bf2-fd2e-4ee0-9692-c0b7736513fb",
    "nics": [
      {
        "subnet_id": "c6bb8788-4fd1-4a8f-adab-7eba3bed8616"
      }
    ],
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,

```

```
    "isAutoRenew": "true",
    "isAutoPay": "true",
    "regionID": "eu-west-101"
  },
  "metadata": {
    "op_svc_userid": "59781460e9e54886a7d03df7d3f3fc81"
  },
  "key_name": "$key_name"
}
```

- **data_volumes.volumetype**: data disk type. **SAS** indicates high I/O.
- **data_volumes.size**: data disk size (GB). The value ranges from **10** to **32768**.
- **data_volumes.shareable**: whether the disk is shareable. **true** indicates a shared disk and **false** indicates a common disk.

Create a BMS with an EIP

You can bind an EIP to the BMS by adding the **publicip** field to the request body. During the BMS creation, an EIP is assigned and bound to the BMS. An example is as follows:

```
{
  "server": {
    "availability_zone": "eu-west-101",
    "name": "bms-test04",
    "imageRef": "766a545a-02e1-433e-b1d1-733b5dc95e94",
    "flavorRef": "physical.s3.large",
    "data_volumes": [
      {
        "volumetype": "SAS",
        "size": 10,
        "shareable": false
      }
    ],
    "vpcid": "8df83bf2-fd2e-4ee0-9692-c0b7736513fb",
    "nics": [
      {
        "subnet_id": "c6bb8788-4fd1-4a8f-adab-7eba3bed8616"
      }
    ],
    "extendparam": {
      "chargingMode": "prePaid",
      "periodType": "month",
      "periodNum": 1,
      "isAutoRenew": "true",
      "isAutoPay": "true",
      "regionID": "eu-west-101"
    },
    "metadata": {
      "op_svc_userid": "59781460e9e54886a7d03df7d3f3fc81"
    },
    "publicip": {
      "eip": {
        "iptype": "5_bgp",
        "bandwidth": {
          "chargemode": "bandwidth",
          "name": "bms-test04-bandwidth",
          "size": 1,
          "sharetype": "PER"
        },
        "extendparam": [
          "chargingMode": "prePaid"
        ]
      }
    },
    "key_name": "$key_name"
  }
}
```

```
}  
}
```

- **eip.ipstype**: EIP type. **5_bgp** indicates dynamic BGP.
- **bandwidth.chargemode**: bandwidth billing type. The value can be **traffic** or **bandwidth**.
- **bandwidth.name**: bandwidth name.
- **bandwidth.size**: bandwidth (Mbit/s). The value ranges from **1** to **2000**.
- **bandwidth.sharetype**: bandwidth sharing type. **PER** indicates dedicated bandwidth.
- **extendparam.chargingMode**: billing mode of the EIP. **prePaid** indicates prepayment, that is, the yearly/monthly mode.

5.2 Example 2: Querying Details About BMSs

Scenarios

You can specify criteria to query details about desired BMSs.

Involved APIs

The following API is involved:

[Querying Details About BMSs](#)

Query Details About BMSs By Specifying a Flavor

- API
URI format: GET /v1/{project_id}/baremetalservers/detail
For details, see [Querying Details About BMSs](#).
- Example request
GET https://{BMS Endpoint}/v1/5233f5df59f9491ba7c14d8b5d535f6b/baremetalservers/detail?flavor=physical.comtest04.large
- Example response

```
{  
  "count": 1,  
  "servers": [  
    {  
      "fault": null,  
      "id": "cb52b43d-8ba7-4b4d-84c4-7ef332715073",  
      "name": "kd1ne4xl-c00420527-nodel-nonstop",  
      "addresses": {  
        "fdbf76aa-697a-4f49-8835-36c0e6903e28": [  
          {  
            "version": "4",  
            "addr": "172.16.0.178",  
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:8a:3a:50",  
            "OS-EXT-IPS:type": "fixed",  
            "OS-EXT-IPS:port_id": "0c111191-9785-49e6-8b42-66ef2802bd8f"  
          },  
          {  
            "version": "4",  
            "addr": "124.71.38.23",  
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:8a:3a:50",  
            "OS-EXT-IPS:type": "floating",  
            "OS-EXT-IPS:port_id": "0c111191-9785-49e6-8b42-66ef2802bd8f"  
          }  
        ]  
      }  
    }  
  ]  
}
```

```
    }
  ]
},
"flavor": {
  "disk": "192010",
  "vcpus": "128",
  "ram": "524288",
  "id": "physical.comtest04.large",
  "name": "physical.comtest04.large"
},
"accessIPv4": "",
"accessIPv6": "",
"status": "SHUTOFF",
"progress": null,
"hostId": "f57b3421b5cdc214fabf127984ef779b7433acc1d6379214ed834219",
"updated": "2020-11-07T09:28:40Z",
"created": "2020-09-21T09:07:28Z",
"metadata": {
  "metering.order_id": "CS20110717284WW42",
  "metering.image_id": "c13c91f7-038a-443e-ad76-608e255b4d85",
  "metering.imagetype": "shared",
  "metering.resourcespeccode": "physical.comtest04.large.linux",
  "image_name": "euler2u8-cbg-sdi3-20200330v3",
  "op_svc_userid": "48b5cec3f26e4bedab54cb996b124079",
  "bms_support_evs": "false",
  "EcmResStatus": "",
  "metering.product_id": "00301-103881-0--0",
  "metering.resourcetype": "__type_baremetal",
  "vpc_id": "fdbf76aa-697a-4f49-8835-36c0e6903e28",
  "os_type": "Linux",
  "os_bit": "64",
  "chargingMode": "1"
},
"tags": [
  "__type_baremetal"
],
"description": "kd1ne4xl-c00420527-nodel-nonstop",
"locked": false,
"config_drive": "",
"tenant_id": "5233f5df59f9491ba7c14d8b5d535f6b",
"user_id": "fc3a5ac3fe244ca6bb079c8069f9143f",
"key_name": null,
"os-extended-volumes:volumes_attached": [
  {
    "device": "/dev/sda",
    "bootIndex": "0",
    "id": "405ae92b-2d59-4d61-8eb0-79946e391c33",
    "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": "f57b3421b5cdc214fabf127984ef779b7433acc1d6379214ed834219",
"OS-EXT-SRV-ATTR:instance_name": "instance-00567827",
"OS-EXT-SRV-ATTR:hypervisor_hostname":
"bea2f2c61b2def48c3ad001446ffada7bbd2b1cbf75d4950c9609a0",
"OS-DCF:diskConfig": "MANUAL",
"OS-EXT-AZ:availability_zone": "eu-west-101",
"os:scheduler_hints": {
  "dec_baremetal": [
    "share"
  ]
},
"OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
"OS-EXT-SRV-ATTR:ramdisk_id": "",
"enterprise_project_id": "0",
"OS-EXT-SRV-ATTR:user_data": "$USER_DATA",
"OS-SRV-USG:launched_at": "2020-09-21T09:14:01.000000",
```

```
{
  "OS-EXT-SRV-ATTR:kernel_id": "",
  "OS-EXT-SRV-ATTR:launch_index": 0,
  "host_status": "UP",
  "OS-EXT-SRV-ATTR:reservation_id": "r-18mfg3us",
  "OS-EXT-SRV-ATTR:hostname": "kd1ne4xl-c00420527-nodel-nonstop",
  "OS-SRV-USG:terminated_at": null,
  "sys_tags": [
    {
      "key": "_sys_enterprise_project_id",
      "value": "0"
    }
  ],
  "security_groups": [
    {
      "id": "2beb2b46-2a55-4b83-b362-20698eeda10b",
      "name": "d310-l00415626-cce-control-8xsdf"
    }
  ],
  "image": {
    "id": "c13c91f7-038a-443e-ad76-608e255b4d85"
  },
  "hypervisor": null
}
```

Query Details About BMSs By Specifying an Enterprise Project ID

- API
URI format: GET /v1/{project_id}/baremetalservers/detail
For details, see [Querying Details About BMSs](#).
- Example request
GET https://{BMS Endpoint}/v1/5233f5df59f9491ba7c14d8b5d535f6b/baremetalservers/detail?enterprise_project_id=98135c62-f0c1-4003-9c9f-7f5523d7f3a2

- Example response

```
{
  "count": 1,
  "servers": [
    {
      "fault": null,
      "id": "cb52b43d-8ba7-4b4d-84c4-7ef332715073",
      "name": "kd1ne4xl-c00420527-nodel-nonstop",
      "addresses": {
        "fdbf76aa-697a-4f49-8835-36c0e6903e28": [
          {
            "version": "4",
            "addr": "172.16.0.178",
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:8a:3a:50",
            "OS-EXT-IPS:type": "fixed",
            "OS-EXT-IPS:port_id": "0c111191-9785-49e6-8b42-66ef2802bd8f"
          },
          {
            "version": "4",
            "addr": "124.71.38.23",
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:8a:3a:50",
            "OS-EXT-IPS:type": "floating",
            "OS-EXT-IPS:port_id": "0c111191-9785-49e6-8b42-66ef2802bd8f"
          }
        ]
      },
      "flavor": {
        "disk": "192010",
        "vcpus": "128",

```

```
"ram": "524288",
"id": "physical.comtest04.large",
"name": "physical.comtest04.large"
},
"accessIPv4": "",
"accessIPv6": "",
"status": "SHUTOFF",
"progress": null,
"hostId": "f57b3421b5cdc214fabf127984ef779b7433acc1d6379214ed834219",
"updated": "2020-11-07T09:28:40Z",
"created": "2020-09-21T09:07:28Z",
"metadata": {
  "metering.order_id": "CS20110717284WW42",
  "metering.image_id": "c13c91f7-038a-443e-ad76-608e255b4d85",
  "metering.imagetype": "shared",
  "metering.resourcespeccode": "physical.comtest04.large.linux",
  "image_name": "euler2u8-cbg-sdi3-20200330v3",
  "op_svc_userid": "48b5cec3f26e4bedab54cb996b124079",
  "__bms_support_evs": "false",
  "EcmResStatus": "",
  "metering.product_id": "00301-103881-0--0",
  "metering.resourcetype": "__type_baremetal",
  "vpc_id": "fdbf76aa-697a-4f49-8835-36c0e6903e28",
  "os_type": "Linux",
  "os_bit": "64",
  "chargingMode": "1"
},
"tags": [
  "__type_baremetal"
],
"description": "kd1ne4xl-c00420527-nodel-nonstop",
"locked": false,
"config_drive": "",
"tenant_id": "5233f5df59f9491ba7c14d8b5d535f6b",
"user_id": "fc3a5ac3fe244ca6bb079c8069f9143f",
"key_name": null,
"os-extended-volumes:volumes_attached": [
  {
    "device": "/dev/sda",
    "bootIndex": "0",
    "id": "405ae92b-2d59-4d61-8eb0-79946e391c33",
    "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": "f57b3421b5cdc214fabf127984ef779b7433acc1d6379214ed834219",
"OS-EXT-SRV-ATTR:instance_name": "instance-00567827",
"OS-EXT-SRV-ATTR:hypervisor_hostname":
"bea2f2c61b2def48c3ad001446ffada47bbd2b1cbf75d4950c9609a0",
"OS-DCF:diskConfig": "MANUAL",
"OS-EXT-AZ:availability_zone": "cn-south-1e",
"os:scheduler_hints": {
  "dec_baremetal": [
    "share"
  ]
},
"OS-EXT-SRV-ATTR:root_device_name": "/dev/vda",
"OS-EXT-SRV-ATTR:ramdisk_id": "",
"enterprise_project_id": "98135c62-f0c1-4003-9c9f-7f5523d7f3a2",
"OS-EXT-SRV-ATTR:user_data": "$USER_DATA",
"OS-SRV-USG:launched_at": "2020-09-21T09:14:01.000000",
"OS-EXT-SRV-ATTR:kernel_id": "",
"OS-EXT-SRV-ATTR:launch_index": 0,
"host_status": "UP",
"OS-EXT-SRV-ATTR:reservation_id": "r-18mfg3us",
"OS-EXT-SRV-ATTR:hostname": "kd1ne4xl-c00420527-nodel-nonstop",
"OS-SRV-USG:terminated_at": null,
```

```
"sys_tags": [
  {
    "key": "_sys_enterprise_project_id",
    "value": "0"
  }
],
"security_groups": [
  {
    "id": "2beb2b46-2a55-4b83-b362-20698eeda10b",
    "name": "d310-l00415626-cce-control-8xsdf"
  }
],
"image": {
  "id": "c13c91f7-038a-443e-ad76-608e255b4d85"
},
"hypervisor": null
}
]
```

5.3 Example 3: Attaching Disks to a BMS

Scenarios

If the existing disks of a BMS fail to meet service requirements, for example, due to insufficient disk space or poor disk performance, you can attach more available disks to the BMS, or call the EVS disk creation API to create disks and attach them to the BMS.

Prerequisites

BMSs using certain flavors cannot have EVS disks attached because the servers do not have SDI iNICs or for other reasons. Therefore, perform the following operations to check whether EVS disks can be attached to a BMS:

Call the [Querying Flavor Details and Extended Flavor Information](#) API and check the response. If the value of **baremetal:__support_evs** for the BMS flavor is **true**, EVS disks can be attached to the BMS. If the value is **false** or the BMS flavor does not contain this parameter, EVS disks cannot be attached to the BMS.

Involved APIs

The following APIs are involved:

- [Creating EVS Disks](#)
- [Attaching an EVS Disk to a BMS](#)
- [Querying EVS disks attached to a BMS](#)

Procedure

1. Create an EVS disk.
 - API
URI format: POST /v2/{project_id}/volumes
For details, see [Creating EVS Disks](#).
 - Example request

POST https://{EVS Endpoint}/v2/05041fffa40025702f6dc009cc6f8f33/volumes

Body:

```
{
  "volume": {
    "name": "openapi_vol01",
    "availability_zone": "eu-west-101",
    "description": "create for api test",
    "volume_type": "SSD",
    "metadata": {
      "hw:passthrough": "true"
    },
    "size": 40
  }
}
```

NOTE

BMSs support only SCSI EVS disks. Therefore, the value of **hw:passthrough** must be **true**.

– Example response

```
{
  "volume": {
    "attachments": [],
    "links": [
      {
        "href": "https://evs.eu-west-101.myhuaweicloud.com/v2/05041fffa40025702f6dc009cc6f8f33/volumes/8ae4ff5b-7e6e-492f-b83a-8a51fdfa3111",
        "rel": "self"
      },
      {
        "href": "https://evs.eu-west-101.myhuaweicloud.com/05041fffa40025702f6dc009cc6f8f33/volumes/8ae4ff5b-7e6e-492f-b83a-8a51fdfa3111",
        "rel": "bookmark"
      }
    ],
    "availability_zone": "eu-west-101",
    "encrypted": false,
    "updated_at": "2020-11-09T08:19:22.077967",
    "replication_status": "disabled",
    "storage_cluster_id": null,
    "snapshot_id": null,
    "id": "8ae4ff5b-7e6e-492f-b83a-8a51fdfa3111",
    "size": 40,
    "user_id": "0504186e6a8010e01f3ec009a7279baa",
    "metadata": {
      "hw:passthrough": "true"
    },
    "status": "creating",
    "description": "create for api test",
    "multiattach": false,
    "source_volid": null,
    "consistencygroup_id": null,
    "name": "openapi_vol01",
    "bootable": "false",
    "created_at": "2020-11-09T08:19:22.061273",
    "volume_type": "SSD",
    "shareable": false
  }
}
```

Record the value of **id** in **volume**.

2. Attach the EVS disk to the BMS.

– API

URI format: POST /v1/{project_id}/baremetalservers/{server_id}/attachvolume

For details, see [Attaching an EVS Disk to a BMS](#).

- Example request

POST https://{BMS Endpoint}/v1/05041fffa40025702f6dc009cc6f8f33/baremetalservers/9ab74d89-61e7-4259-8546-465fdebe4944/attachvolume

Body:

```
{
  "volumeAttachment": {
    "volumeId": "8ae4ff5b-7e6e-492f-b83a-8a51fdfa3111",
    "device": "/dev/sdb"
  }
}
```

- Example response

```
{
  "volumeAttachment": {
    "id": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "volumeId": "8ae4ff5b-7e6e-492f-b83a-8a51fdfa3111",
    "serverId": "9ab74d89-61e7-4259-8546-465fdebe4944",
    "device": "/dev/sdb"
  }
}
```

3. Check whether the EVS disk is successfully attached to the BMS.

- API

URI format: GET /v1/{project_id}/baremetalservers/{server_id}/os-volume_attachments

For details, see [Querying EVS Disks Attached to a BMS](#).

- Example request

GET https://{BMS Endpoint}/v1/05041fffa40025702f6dc009cc6f8f33/baremetalservers/9ab74d89-61e7-4259-8546-465fdebe4944/os-volume_attachments

- Example response

```
{
  "volumeAttachments": [
    {
      "device": "/dev/sdd",
      "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
      "serverId": "9ab74d89-61e7-4259-8546-465fdebe4944",
      "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
    },
    {
      "device": "/dev/sdb",
      "id": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
      "serverId": "9ab74d89-61e7-4259-8546-465fdebe4944",
      "volumeId": "8ae4ff5b-7e6e-492f-b83a-8a51fdfa3111"
    }
  ]
}
```

6 Public Parameters

6.1 Status Codes

- Normal

Returned Value	Description
200 OK	The request has been successfully processed.
201 Created	The request has been successfully processed and a resource has been created.
202 Accepted	The request has been accepted but may not be processed immediately.
204 No Content	The request has been successfully processed, but there is no need to send any data back.

- Abnormal

Returned Value	Description
400 Bad Request	The request failed to be processed due to bad syntax.
401 Unauthorized	A username and a password are required to access a page.
403 Forbidden	The requested page cannot be accessed.
404 Not Found	The requested page cannot be found.
405 Method Not Allowed	A method in the request is not allowed.
406 Not Acceptable	The response cannot be accepted by the client.
407 Proxy Authentication Required	The client must be authorised by the proxy before the request can proceed.

Returned Value	Description
408 Request Timeout	The request timed out.
409 Conflict	Failed to complete the request due to a conflict.
500 Internal Server Error	Failed to complete the request due to a system error.
501 Not Implemented	Failed to complete the request because the server does not support all that is needed for the request to be completed.
502 Bad Gateway	Failed to complete the request because the server has received an invalid response from an upstream server.
503 Service Unavailable	Failed to complete the request because the server is unavailable.
504 Gateway Timeout	The server has waited too long for a response from an upstream server.

6.2 Error Codes

Context

- An error code returned by an API does not correspond to one error message. The following table lists only common error messages.
- Most BMS APIs are asynchronous. Some error codes are displayed in the returned messages for task viewing requests. HTTP status codes may not be accurate.
- The BMS service is strongly dependent on other services, such as network and storage. When error messages are provided for the BMS-depended services, contact technical support for troubleshooting.

Error Code Description

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](#).

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0001	Fail to parse request, reason: %s.	Request error.	Check the request body according to the returned error message.

Statu s Code	Error Code	Error Message	Description	Handling Measure
400	BMS.00 02	The BMS service is unavailable.	BMS has not been launched.	Select a region where the BMS service is available.
400	BMS.00 03	Create BareMetal Server error, request is null.	Request error.	Check the request body according to the returned error message.
400	BMS.00 04	Role check fail, reason: %s.	Insufficient permission.	See the returned error message or contact technical support.
400	BMS.00 05	Role check fail, reason: You do not have permission or your balance is insufficient.	Insufficient permission.	See the returned error message or contact technical support.
400	BMS.00 08	Fail to query flavor [%s], reason: %s.	Failed to query a flavor.	See the returned error message or contact technical support.
400	BMS.00 06	Fail to operate baremetal server.	Failed to create the task.	See the returned error message or contact technical support.
400	BMS.00 09	Fail to query flavor extra specs[%s], reason: %s.	Failed to query flavor attributes.	See the returned error message or contact technical support.
400	BMS.00 10	Fail to query image [%s], reason: %s.	Failed to query an image.	See the returned error message or contact technical support.
400	BMS.00 11	The selected images cannot be used to apply for BMSs, %s.	The images do not support BMS.	See the returned error message or contact technical support.
400	BMS.00 12	The number of IP addresses in the selected subnet[%s] is insufficient.	Insufficient IP addresses in the selected subnet.	Check whether the IP addresses of the subnet are used up.
400	BMS.00 13	Fail to query ports by subnet [%s], reason: %s.	Failed to query the port.	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0014	The specified IP address conflicts with an existing IP address in subnet[%s].	IP address conflict.	Modify the NIC IP address.
400	BMS.0015	Fail to query subnet, reason: %s.	Failed to query the NIC.	See the returned error message or contact technical support.
400	BMS.0017	The number[%d] of EIPs has reached the maximum[%d] allowed. Apply for a higher quota and try again.	The EIP quota is insufficient.	Apply for a higher EIP quota.
400	BMS.0218	Fail to create order, reason: %s.	Failed to create the order.	See the returned error message or contact technical support.
400	BMS.0219	OpenStack nova change ServerName fail.	Failed to change the BMS name.	Contact technical support.
400	BMS.0018	Request parameter is invalid.	Invalid request parameters.	Modify the request parameters based on the returned error message.
400	BMS.0019	publicIp parameter is illegal, reason: %s.	Invalid NIC parameters.	See the returned error message or contact technical support.
400	BMS.0020	Fail to check chargingMode, reason: %s.	Failed to check the billing mode.	See the returned error message or contact technical support.
400	BMS.0021	Flavor parameter is illegal, reason: %s.	The flavor is invalid for creating the BMS. Select another flavor.	Select a valid flavor.
400	BMS.0022	Request parameter is invalid.	Invalid request parameters.	Modify the request parameters based on the returned error message.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0023	Fail to query limits, reason: %s	Failed to query the quota.	See the returned error message or contact technical support.
400	BMS.0025	The number of cloud servers has reached the maximum allowed.	The number of BMSs exceeds the quota.	Apply for a higher quota.
400	BMS.0026	The number of CPU cores used by all cloud servers has reached the maximum allowed.	The number of BMS CPUs exceeds the quota.	Apply for a higher quota.
400	BMS.0027	The memory space used by all cloud servers has reached the maximum allowed.	The BMS memory exceeds the quota.	Apply for a higher quota.
400	BMS.0028	The number of CPU cores used by all cloud servers and that of cloud servers have reached the maximum allowed.	The number of BMSs and that of CPUs exceed the quotas.	Apply for a higher quota.
400	BMS.0029	The memory space used by all cloud servers and the number of cloud servers have reached the maximum allowed.	The number of BMSs and the memory exceed the quotas.	Apply for a higher quota.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0030	The memory space and number of CPU cores used by all cloud servers have reached the maximum allowed.	The BMS memory and the number of CPUs exceed the quotas.	Apply for a higher quota.
400	BMS.0031	The number of cloud servers, the memory space used by all cloud servers, and the number of CPU cores used by all cloud servers have reached the maximum allowed.	The number of BMSs, the number of CPUs, and the memory exceed the quotas.	Apply for a higher quota.
400	BMS.0032	Token check fail.	Contact technical support to apply for a BMS.	See the returned error message or contact technical support.
403	BMS.0033	Fail to check roles, reason: %s.	You do not have operation rights. Contact technical support.	See the returned error message or contact technical support.
400	BMS.0034	Not support create Bare Metal Server.	Currently, BMSs cannot be automatically provisioned.	This operation is not supported.
400	BMS.0047	Root volume is illegal, reason: %s.	Invalid system disk.	See the returned error message or contact technical support.
400	BMS.0049	Query keypair fail, reason is: %s.	Failed to query key_name .	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0089	The tag policy does not allow the specified value for the following tag key: %s.	Failed to pass the verification of tag policy compliance.	Change the tag values according to the policy or contact technical support.
400	BMS.0102	The image does not support attach volume.	This image does not support volume attaching.	Contact technical support.
400	BMS.0103	Attach volume %s fail, volume info is null.	The disk does not exist.	Check whether the disk information is correct.
400	BMS.0104	Attach volume %s fail, data volume can not attach in root volume device.	The data volume cannot be attached to the mount point of the system volume.	Select another mount point.
400	BMS.0105	Attach volume %s fail, device is illegal.	The mount point is invalid.	Check whether the mount point is valid.
400	BMS.0106	Attach volume %s fail, volume's az is not equal with server's az.	The AZ of data disks is different from that of the BMS.	Ensure that the AZ of data disks is the same as that of the BMS.
400	BMS.0108	Attach volume %s fail, root volume does not match the VM, do not change image.	The system disk does not match the BMS.	Attach a matching system disk to the BMS.
400	BMS.1001	Volume can only be attached when server %s stopped or active.	This operation can be performed only when the BMS is in running or stopped.	Try again later or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.1002	Attach volume fail, server %s attachment num over limit.	The number of data disks that can be attached to the BMS exceeds the quota.	Contact technical support.
400	BMS.1003	Attach shareable volume %s fail, volume status is %s, not available or inuse.	The disk status is invalid.	Check whether the disk is in attached or other unavailable status.
400	BMS.1004	Attach shareable volume %s fail, volume status is %s, not available or inuse.	The number of shared data disks that can be attached to the BMS exceeds the quota.	Contact technical support.
400	BMS.1006	Server %s info is null or its status or its metadata is null.	The running status of the BMS cannot be obtained.	Try again later or contact technical support.
400	BMS.1007	Calling interface failed	Failed to call the API.	See the returned error message or contact technical support.
400	BMS.1008	The server[%s] is not HANA server, volume[%s] does not match the server.	The disk type does not match the BMS.	Check whether the disk type matches the BMS or contact the technical support.
400	BMS.1009	Attach shareable volume %s fail, volume status is %s, the volume has been attached to the server.	The data disk has been attached to the BMS.	Select another data disk.
400	BMS.1011	Attach shareable volume %s fail, volume status is %s, not available or inuse.	The data disk is being attached.	Do not repeatedly attach the data disk.

Statu s Code	Error Code	Error Message	Description	Handling Measure
400	BMS.30 01	Query security group failed: %s. or Query vpclId for subnet failed: %s.	Failed to query the security group or subnet.	See the returned error message or contact technical support.
400	BMS.30 11	Decoded token is null.	System error.	See the returned error message or contact technical support.
400	BMS.30 25	Not support create shareable data volumes.	The EVS disk type is incorrect.	See the returned error message or contact technical support.
400	BMS.30 35	Attach volume fail: the bmsid [%s] in volume metadata is not the same with bmsid [%s] in url.	No matching disk is found for the BMS.	Ensure that the disk matches the BMS or contact technical support.
400	BMS.30 39	The password is illegal, reason: %s.	Failed to verify password complexity.	See the password rules.
400	BMS.02 46	The vpclId[%s] is invalid or not- existing.	Failed to query the VPC.	See the returned error message or contact technical support.
400	BMS.02 01	Fail to check the baremetal server params, reason: %s.	The parameters for creating the BMS are incorrectly configured.	See the returned error message or contact technical support.
400	BMS.02 02	All volumes must be in same type.	The data disk type is invalid.	See the returned error message or contact technical support.
400	BMS.02 03	Volume is illegal, %s.	Parameter Volume is invalid.	See the returned error message or contact technical support.
400	BMS.02 04	The num of datavolume is illegal, the num is %d, max num allow is %d.	The number of data disks exceeds the upper limit.	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0205	The number of baremetal servers is out of range for one quest.	The BMS quantity is invalid.	Change or delete the BMS quantity.
400	BMS.0206	The baremetal server name [%s] is include invalid char.	The name contains invalid characters.	Change the name as prompted.
400	BMS.0207	VPC is illegal, reason: %s.	No IP address can be specified when BMSs are created in a batch.	See the returned error message or contact technical support.
400	BMS.0208	Fail to get RegionInfo by tenant [%s], reason: %s.	Failed to query the AZ.	See the returned error message or contact technical support.
400	BMS.0210	The size of userdata [%d] is over quota limits [%d].	The length of the injected data exceeds the upper limit.	Modify the injected data.
400	BMS.0211	The image platform is [%s], support publicKey, reason: the publicKey is illegal, null is not allowed.	Invalid key.	Select a valid key.
400	BMS.0114	Volume %s is not in server %s attach volume list.	The disk to be detached is not in the disk list of the BMS.	Check whether the disk exists.
400	BMS.0212	System is overloaded, please try again later.	The system is overloaded.	Try again later.
400	BMS.0213	OBTAZ role verify fail: not allowed role.	Insufficient permission.	See the returned error message or contact technical support.

Statu s Code	Error Code	Error Message	Description	Handling Measure
400	BMS.02 14	Query subnet[%s] failed: response is null.	Failed to query the NIC.	See the returned error message or contact technical support.
400	BMS.02 15	The response of inquiry order info is null or invalid.	Failed to create the order.	See the returned error message or contact technical support.
400	BMS.02 16	Submit order[%s] failed: %s.	Failed to submit the order.	See the returned error message or contact technical support.
400	BMS.02 17	Create server fail, reason: metaData is illegal.	Failed to verify metadata.	See the returned error message or contact technical support.
400	BMS.02 22	primary port can not be deleted.	The primary NIC cannot be deleted.	See the returned error message or contact technical support.
400	BMS.02 23	Only SCSI disks are supported.	Currently, only SCSI disks are supported.	Select the SCSI disk type.
400	BMS.00 39	Server is stopped, not allow to stop.	The BMS has been shut down.	Do not stop the BMS again.
400	BMS.00 40	The server does not exist.	The BMS does not exist.	See the returned error message or contact technical support.
400	BMS.00 35	Server is powering on, not allow to %s.	The BMS is being powered on and no other operations are allowed.	See the returned error message or contact technical support.
400	BMS.00 36	Server is powering off, not allow to %s.	The BMS is being powered off and no other operations are allowed.	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0038	Server is running, not allow to start.	The BMS is running and cannot be started.	Do not start the BMS again.
400	BMS.0037	Server is rebooting, not allow to %s.	The BMS is being restarted and no other operations are allowed.	See the returned error message or contact technical support.
403	BMS.0288	Policy doesn't allow %s to be performed.	You do not have permissions to perform this operation.	See the returned error message or contact technical support.
400	BMS.0290	Batch attach volume type must be sharable.	The disk is not a shared disk and cannot be attached to multiple BMSs.	Select a shared disk.
400	BMS.0291	enterprise_project_id can not be empty.	The enterprise project ID cannot be empty.	See the returned error message or contact technical support.
400	BMS.0294	Query enterprise_project_id[%s] failed %.	Failed to invoke EPS to query the project.	See the returned error message or contact technical support.
400	BMS.0293	Query enterprise_project_id[%s] failed: response is null.	Failed to query the project information.	See the returned error message or contact technical support.
400	BMS.0292	enterprise_project_id[%s] is deactive.	The enterprise project has been suspended.	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0071	Using an existing EIP and creating a new EIP can't be used at the same time.	If an EIP has been specified for creating the BMS, no other EIP can be created for the BMS.	See the returned error message or contact technical support.
400	BMS.0072	An existing EIP cannot be assigned to the ECSs created in batches.	An EIP cannot be used to create multiple BMSs.	See the returned error message or contact technical support.
400	BMS.0073	Bandwidth info in eip is null.	The bandwidth parameter is empty.	See the returned error message or contact technical support.
400	BMS.0074	Parameter exetendparam or chargingMode is null.	The EIP extension parameter is unavailable.	See the returned error message or contact technical support.
400	BMS.0075	Bandwidth info in eip is null.	The bandwidth ID is empty.	See the returned error message or contact technical support.
400	BMS.0076	Parameter chargingMode only supports postPaid when shared type is whole.	The EIP billing mode is unavailable.	See the returned error message or contact technical support.
400	BMS.0077	PublicIp parameter is illegal, reason: %s.	Invalid bandwidth size.	See the returned error message or contact technical support.
400	BMS.0078	Shared bandwidth has been bound to %d EIPs, quota is %d.	EIP quota is insufficient.	See the returned error message or contact technical support.
400	BMS.0079	Specifies the bandwidth of the shared type and the id cannot be empty.	The EIP bandwidth type is unavailable.	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.0080	Check subnet status failed.	The subnet status is unavailable.	Check whether the subnet exists or whether it is in ACTIVE state.
400	BMS.0297	Fail to attach volume[%s]: volume is locked.	The submitted EVS disk order has not been paid. The EVS disk cannot be attached to or detached from the BMS.	See the returned error message or contact technical support.
400	BMS.0054	Check capacity fail, the number of capacity is: [%d], and req num is: [%d].	Failed to verify the capacity.	The capacity is insufficient. Contact technical support.
400	BMS.0055	Query capacity fail, Flavor id is [%s], reason: [%s].	Failed to query the quota.	See the returned error message or contact technical support.
400	BMS.2005	Network could not be found.	Failed to query the network.	Check the network information or contact technical support.
400	BMS.3037	Insufficient resources or failed to start the BMS.	Insufficient resources or failed to start the BMS.	Contact technical support.
400	BMS.3004	Failed to create the BMS due to an internal system error.	Failed to create the BMS due to an internal system error.	Contact technical support.
400	BMS.3005	Failed to create the port.	Failed to create the port.	Contact technical support.
400	BMS.3033	Failed to create the system disk.	Failed to create the system disk.	Contact technical support.

Statu s Code	Error Code	Error Message	Description	Handling Measure
400	BMS.30 29	Failed to create the system disk. The disk status is abnormal.	Failed to create the system disk. The disk status is abnormal.	Contact technical support.
400	BMS.30 06	Failed to assign the floating IP address.	Failed to assign the EIP.	Contact technical support.
400	BMS.30 21	Failed to create the data disk.	Failed to create the data disk.	Contact technical support.
400	BMS.30 19	Failed to attach the data disk.	Failed to attach the data disk.	Contact technical support.
400	BMS.30 38	Failed to assign the EIP.	Failed to assign an EIP.	Contact technical support.
400	BMS.03 15	Disks cannot be attached to a BMS using this flavor[%s].	Disks cannot be attached to a BMS using this flavor.	See the returned error message or contact technical support.
400	BMS.03 16	RAID level and RAID type do not match.	The RAID level does not match the RAID type.	See the returned error message or contact technical support.
400	BMS.03 17	Disk information is empty.	Disk information is empty.	See the returned error message or contact technical support.
400	BMS.03 20	This flavor does not support RAID formed by disks of this type.	This flavor does not support RAID formed by disks of this type.	See the returned error message or contact technical support.
400	BMS.03 21	This flavor does not support RAID formed by disks of this size.	This flavor does not support RAID formed by disks of this size.	See the returned error message or contact technical support.

Statu s Code	Error Code	Error Message	Description	Handling Measure
400	BMS.03 22	Disk quantity and RAID level do not match.	Disk quantity and RAID level do not match.	See the returned error message or contact technical support.
400	BMS.03 23	Disk quantity does not match that in the flavor.	Disk quantity does not match that in the flavor.	See the returned error message or contact technical support.
400	BMS.03 25	Disk quantity exceeds that supported by the flavor.	Disk quantity exceeds that supported by the flavor.	See the returned error message or contact technical support.
400	BMS.03 27	Only JBOD is supported.	Only JBOD is supported.	See the returned error message or contact technical support.
400	BMS.03 28	Only single-disk RAID 0 is supported.	Only single-disk RAID 0 is supported.	See the returned error message or contact technical support.
400	BMS.03 29	Disk type is empty.	Disk type is empty.	See the returned error message or contact technical support.
400	BMS.03 30	RAID is not supported.	RAID is not supported.	See the returned error message or contact technical support.
400	BMS.03 31	No IPv6 Open Beta Test permissions.	No IPv6 Open Beta Test permissions.	See the returned error message or contact technical support.
400	BMS.03 33	IPv6 is not supported.	IPv6 is not supported.	See the returned error message or contact technical support.
400	BMS.04 00	the image[%s] is not support IPv6.	The image does not support IPv6.	See the returned error message or contact technical support.
400	BMS.03 60	This operation cannot be performed because EVS Disk %s is frozen.	The disk has been frozen and cannot be operated for the BMS.	See the returned error message or contact technical support.

Status Code	Error Code	Error Message	Description	Handling Measure
400	BMS.3026	Password does not meet the requirements of the rule.	The password does not meet requirements.	See the returned error message or contact technical support.
400	BMS.3027	VPC parameter is illegal.	Invalid VPC parameters.	See the returned error message or contact technical support.
400	BMS.3028	cinder quota check fail.	Disk quota is abnormal.	See the returned error message or contact technical support.
400	Common.0018	tenantId in token is not the same with in URL.	tenant_id in the token is different from that in the URL.	Check whether the tenant token is correct.
400	BMS.0111	none or multiple passwords specified, please specify one.	Password or key pair is not specified, or both specified.	The password and key pair cannot be specified at the same time. Select one of them based on the login mode.
400	BMS.0395	The server[%s] is windows system and can not support remote login.	Windows BMSs do not support remote login.	See the returned error message or contact technical support.
400	BMS.0605	the BMS is locked.	The BMS is locked.	See the returned error message or contact technical support.

6.3 Task Responses

6.3.1 Task ID Response

Normal Response

Table 6-1 Normal response

Parameter	Type	Description
job_id	String	Specifies the task ID after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

Abnormal Response

Table 6-2 Abnormal response

Parameter	Type	Description
error	Dictionary data structure	Specifies the error returned when a task submission encounters an exception. For details, see Table 6-3 .

Table 6-3 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

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

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

6.3.2 Order ID Response

Normal Response

Table 6-4 Normal response

Parameter	Type	Description
order_id	String	Specifies the order ID returned after an order is submitted. You can query the order processing progress based on the ID. To query the order processing progress, log in to the management console and choose Billing > My Orders .
job_id	String	Specifies the task ID returned after a task command is issued. The task ID can be used to query the execution status of the task. For details about how to query the task execution status based on job_id , see Querying Task Statuses .

Abnormal Response

Table 6-5 Abnormal response

Parameter	Type	Description
error	Dictionary data structure [1]	Specifies the error returned when a task submission encounters an exception. For details, see Table 6-6 .

Table 6-6 error data structure

Parameter	Type	Description
message	String	Specifies the error message.
code	String	Specifies the error code.

Example Response

- Normal response

```
{
  "order_id": "CS2009141523OQSEQ",
  "job_id": "ff808081748b760c01748b7f80370003"
}
```
- Abnormal response

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

7 Permissions and Supported Actions

7.1 Introduction

You can use Identity and Access Management (IAM) for fine-grained permissions management of your BMSs. If your Huawei Cloud account does not need individual IAM users, you can skip this section.

By default, new IAM users do not have any permissions granted. You need to add a user to one or more groups, and assign policies or roles to these groups. The user then inherits permissions from the groups it is a member of. This process is called authorization. After authorization, the user can perform specified operations on BMS based on the permissions.

Roles are provided by IAM to define service-based permissions depending on users' job responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

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

An account has all the permissions required to call any API, but IAM users must have permissions assigned to call a specific API. The permissions required for calling an API are determined by the action supported by the API. Only users with the policies allowing for such an action can call the API. For example, if an IAM user wants to call an API to query BMSs, the user must be granted the permissions allowing for the **bms:servers:list** action.

Supported Actions

BMS provides system-defined policies that can be directly used in IAM. You can also create custom policies and use them to supplement system-defined policies, implementing more refined access control. Actions supported by policies are specific to APIs. The following are common concepts related to policies:

- Permission: a statement in a policy that allows or denies certain operations.
- APIs: REST APIs that can be called in a custom policy.

- Actions: Added to a custom policy to control permissions for specific operations.
- Dependent actions: When assigning an action to users, you also need to assign dependent permissions for that action to take effect.
- IAM project/Enterprise project: A custom policy can be applied to IAM projects or enterprise projects or both. Policies that contain actions supporting both IAM and enterprise projects can be assigned to user groups and take effect in both IAM and Enterprise Management. Policies that only contain actions supporting IAM projects can be assigned to user groups and only take effect for IAM. Such policies will not take effect if they are assigned to user groups in Enterprise Management.

 NOTE

√: supported; x: not supported

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

- **Lifecycle management actions**, including actions supported by BMS's lifecycle management APIs, such as the APIs for creating BMSs and querying BMS details.
- **Status management actions**, including actions supported by BMS's status management APIs, such as the APIs for stopping, restarting, and starting BMSs in batches.
- **Password management actions**, including actions supported by BMS's password-related APIs, such as the APIs for querying whether BMS passwords can be reset.
- **Flavor query actions**, including actions supported by BMS's flavor-related APIs, such as the APIs for querying details about flavors and extended flavor information.
- **NIC management actions**, including actions supported by BMS's NIC-related APIs, such as the APIs for querying BMS NICs.
- **Disk management actions**, including actions supported by BMS's disk-related APIs, such as the APIs for attaching disks to a BMS and detaching disks from a BMS.
- **Metadata management actions**, including actions supported by BMS's metadata-related APIs, such as the API for updating the BMS metadata.
- **Tenant quota management actions**, including actions supported by BMS's quota-related APIs, such as the API for querying tenant quotas.

Action Descriptions

- The actions and scopes of native OpenStack APIs of BMS are the same as those of ECS. For details, see in *Elastic Cloud Server API Reference*.
- When customizing a BMS user policy in IAM, add the `ecs:*:get` and `ecs:*:list` permissions. Otherwise, functions on some pages cannot work properly.
- High-speed networks and user-defined networks do not support enterprise projects. To use the two functions, use the primary account.
- The `vpc:ports:get` permission must be assigned to the sub-account of an enterprise project. Otherwise, the EIP and security group cannot be properly displayed on the BMS details page.

7.2 Lifecycle Management

Permissions	API	Action	IAM Project	Enterprise Project
Creating a BMS	POST /v1/{project_id}/baremetalservers	bms:servers:create	√	√
Querying Details About BMSs	GET /v1/{project_id}/baremetalservers/detail	bms:servers:list	√	√
Querying Details About a BMS	GET /v1/{project_id}/baremetalservers/{server_id}	bms:servers:get	√	√

7.3 Status Management

Permissions	API	Action	IAM Project	Enterprise Project
Change the Name of a BMS	PUT /v1/{project_id}/baremetalservers/{server_id}	bms:servers:put	√	√
Reinstalling the BMS OS	POST /v1/{project_id}/baremetalservers/{server_id}/reinstallos	bms:servers:reinstallOS	√	√
Stopping BMSs in a Batch	POST /v1/{project_id}/baremetalservers/action	bms:servers:stop	√	√
Restarting BMSs in a Batch	POST /v1/{project_id}/baremetalservers/action	bms:servers:reboot	√	√
Starting BMSs in a Batch	POST /v1/{project_id}/baremetalservers/action	bms:servers:start	√	√

7.4 Flavor Query

Permissions	API	Action	IAM Project	Enterprise Project
Querying Details About Flavors and Extended Flavor Information	GET /v1/{project_id}/baremetalservers/flavors	bms:serverFlavors:get	√	√

7.5 NIC Management

Permissions	API	Action	IAM Project	Enterprise Project
Querying Information About BMS NICs	GET /v1/{project_id}/baremetalservers/{server_id}/os-interface	bms:servers:get	√	√

7.6 Disk Management

Permissions	API	Action	Dependent Actions	IAM Project	Enterprise Project
Detaching a Disk from a BMS	DELETE /v1/{project_id}/baremetalservers/{server_id}/detachvolume/{attachment_id}	bms:servers:detachVolume	-	√	√
Attaching a Disk to a BMS	POST /v1/{project_id}/baremetalservers/{server_id}/attachvolume	bms:servers:attachVolume	evs:volume:use	√	√

Permissions	API	Action	Dependent Actions	IAM Project	Enterprise Project
Querying EVS disks attached to a BMS	GET /v1/{project_id}/baremetalservers/{server_id}/os-volume_attachments	bms:servers:get	-	√	√

7.7 Metadata Management

Permissions	APIs	Actions	IAM Project	Enterprise Project
Updating the Metadata of a BMS	POST /v1/{project_id}/baremetalservers/{server_id}/metadata	bms:servers:updateMetadata	√	√

7.8 Tenant Quota Management

Permissions	APIs	Actions	IAM Project	Enterprise Project
Querying the Tenant Quota	GET /v1/{project_id}/baremetalservers/limits	bms:serverQuotas:get	√	x

7.9 Password Management

Permissions	API	Action	IAM Project	Enterprise Project
Querying Whether a BMS Supports Password Reset	GET /v1/{project_id}/baremetalservers/{server_id}/os-resetpwd-flag	bms:servers:get	√	√

Permissions	API	Action	IAM Project	Enterprise Project
Obtaining a password from a Windows BMS	GET /v1/{project_id}/baremetalservers/{server_id}/os-server-password	bms:server: s:get	√	√
Deleting the password of a Windows BMS	DELETE /v1/{project_id}/baremetalservers/{server_id}/os-server-password	bms:server: s:deletePas sword	√	√

8 Historical APIs

8.1 API Instructions (OpenStack Nova APIs)

- For details about network service APIs, see [Virtual Private Cloud API Reference](#).
- For the native OpenStack APIs not provided in this document (such as the API used to attach EVS disks to a BMS), use the native OpenStack APIs of ECS. For details, see [Elastic Cloud Server API Reference](#).
- 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.

8.2 API Version Query (OpenStack Nova APIs)

8.2.1 Querying API Versions

Function

This API is used to query all available Nova versions.

URI

GET /

Request Parameters

None

Example Request

Querying all available API versions
GET https://{ECS Endpoint}/

Response Parameters

Parameter	Type	Description
versions	Array of objects	Specifies the list of all API versions.
id	String	Specifies the version ID, for example, v1.
links	Array of objects	Specifies the API URL.
href	String	Specifies the reference address of the current API version.
rel	String	Specifies the relationship between the current API version and the referenced address.
version	String	If the APIs of this version support minor versions, set this parameter to the maximum minor version supported. If not, leave this parameter blank.
status	String	Specifies the version status. Possible values are as follows: ● CURRENT: indicates a primary version. ● SUPPORTED: indicates an old version that is still supported. ● DEPRECATED: indicates a deprecated version which may be deleted later.
updated	String	Specifies the version release time, which must be the UTC time. For example, the release time of v1 is 2014-06-28T12:20:21Z.
min_version	String	If the APIs of this version support minor versions, set this parameter to the supported minimum minor version. If not, leave this parameter blank.

Example response

```
{  
  "versions": [  

```

```
{
  "links": [
    {
      "rel": "self",
      "href": "https://192.168.82.231:443/v2/"
    }
  ],
  "id": "v2.0",
  "updated": "2018-09-21T12:33:21Z",
  "status": "SUPPORTED"
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.2.2 Querying an API Version

Function

This API is used to query a specified API version.

URI

GET /{api_version}

[Table 8-1](#) lists the parameters.

Table 8-1 Parameter description

Parameter	Mandatory	Description
api_version	Yes	Specifies the API version, for example v2.

Request Parameters

None

Example Request

Querying the API v2
GET https://{ECS Endpoint}/v2

Response Parameters

Parameter	Type	Description
version	Object	Specifies a specified version.
id	String	Specifies the version ID, for example, v1.
links	Array of objects	Specifies the API URL.
href	String	Specifies the reference address of the current API version.
rel	String	Specifies the relationship between the current API version and the referenced address.
version	String	If the APIs of this version support minor versions, set this parameter to the maximum minor version supported. If not, leave this parameter blank.
status	String	Specifies the version status. Possible values are as follows: • CURRENT: indicates a primary version. • SUPPORTED: indicates an old version that is still supported. • DEPRECATED: indicates a deprecated version which may be deleted later.
updated	String	Specifies the version release time, which must be the UTC time. For example, the release time of v1 is 2014-06-28T12:20:21Z.
min_version	String	If the APIs of this version support minor versions, set this parameter to the supported minimum minor version. If not, leave this parameter blank.

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

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.3 BMS Lifecycle Management (OpenStack Nova APIs)

8.3.1 Creating a BMS (Deprecated)

Function

This API is used to create a BMS.

This API has been deprecated. Use the API in [Creating BMSs](#).

Constraints

- This API cannot be used to create BMSs in a batch.
- When you create a BMS using an image that supports Cloud-Init or Cloudbase-Init, only parameter **key_name** can be configured. (Parameter **adminPass** is invalid.) The password of a Linux BMS can be injected only using parameter **user_data**. The password of a Windows BMS can be injected only using metadata **admin_pass**.
- When you create a BMS using an image that does not support Cloud-Init or Cloudbase-Init, both parameters **adminPass** and **key_name** are invalid. You need to use the password or certificate of the image to log in to the BMS.

- File injection is not supported.
- BMS creation from a system volume is not supported.
- 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.
- After a BMS is created, it is recommended that you add the **__type_baremetal** tag for it. This tag specifies that the created server is a BMS. Otherwise, the BMS may not be displayed in the BMS list on the management console.
- A BMS can have a maximum of two VPCs. The first VPC will be used by the primary NIC of the BMS.

URI

POST /v2.1/{project_id}/servers

[Table 8-2](#) lists the parameters.

Table 8-2 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID.

Request Parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the BMS details. For details, see Table 8-3 .

Table 8-3 server data structure

Parameter	Mandatory	Type	Description
imageRef	Yes	String	Specifies the ID or URL of the image used by the BMS. • Example image ID: 3b8d6fef-af77-42ab-b8b7-5a7f0f0af8f2 • Example image URL: http://glance.openstack.example.com/images/3b8d6fef-af77-42ab-b8b7-5a7f0f0af8f2 NOTE • BMSs using certain flavors do not support all public images provided by the cloud service platform. To obtain the images supported by a BMS flavor, log in to the management console, view the images displayed on the Create BMS page, and obtain the image IDs on the Image Management Service page. • If the creation fails, modify the parameter settings.
flavorRef	Yes	String	Specifies the ID or URL of the flavor used by the BMS.
name	Yes	String	Specifies the BMS name. It contains fewer than 256 characters and cannot be left blank.
metadata	No	Object	Specifies the BMS metadata. The maximum length for both the metadata key and value is 255 characters. For details, see Table 8-4.

Parameter	Mandatory	Type	Description
user_data	No	String	Specifies the user data to be injected during the BMS creation. Text, text files, and .gzip files can be injected. The content to be injected cannot be greater than 32 KB in size. The content to be injected must be encoded with base64.

Parameter	Mandatory	Type	Description
adminPass	No	String	Specifies the initial login password of the BMS administrator account. This parameter is invalid for a Linux BMS. The administrator account of a Windows BMS is Administrator. Password complexity requirements: • The password contains 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 BMS password cannot contain the username, the username in reverse order, or more than two consecutive characters in the username. Note: If this parameter is not specified, a random password will be generated. Special characters: !@\$%^_-=+[]{};.,/?
security_groups	No	Array of objects	Specifies the security group of a BMS. The default value is default. This parameter is valid when you specify parameter network. You are not allowed to specify multiple security groups. For details, see Table 8-5.

Parameter	Mandatory	Type	Description
networks	Yes	Array of objects	Specifies the BMS NICs. For details, see Table 8-6 . You can specify a maximum of four networks for a BMS, including two VXLAN networks and two GENEVE networks. The first network in the parameter must be a VXLAN network. The network is used as by the primary NIC of the BMS. If multiple groups of network parameters are specified, ensure that the parameters of each group belong to the same VPC.
key_name	No	String	Specifies the name of a key pair. This is an extended attribute.
availability_zone	Yes	String	Specifies information about the AZ that the BMS belongs to. You are not allowed to specify host information.

Table 8-4 metadata data structure

Parameter	Mandatory	Type	Description
User-defined field key and value pair	No	String	Specifies the key and value pair of the metadata. Each key or value contains a maximum of 255 characters.

Table 8-5 security_groups data structure

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the name of the security group that the BMS belongs to.

Table 8-6 networks data structure

Parameter	Mandatory	Type	Description
port	No	String	Specifies the UUID of the network port.
uuid	No	String	Specifies the network UUID.
fixed_ip	No	String	Specifies the fixed IP address.

Example Request

Creating a BMS (image ID: 1a6635d8-afea-4f2b-abb6-27a202bad319; flavor: physical.o2.medium; name: bms_name01; AZ: az-dc-1; network UUID: 8470310b-bfa2-4edf-8f64-d15196b2b2c9)

POST https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers

```
{
  "server": {
    "imageRef": "1a6635d8-afea-4f2b-abb6-27a202bad319",
    "flavorRef": "physical.o2.medium",
    "name": "bms_name01",
    "availability_zone": "az-dc-1",
    "networks": [
      {
        "uuid": "8470310b-bfa2-4edf-8f64-d15196b2b2c9"
      }
    ]
  }
}
```

Response Parameters

Parameter	Type	Description
server	Object	Specifies BMS details. For details, see Table 8-7 .

Table 8-7 server data structure

Parameter	Type	Description
security_groups	Array of objects	Specifies details about the security group of the BMS. For details, see Table 8-8 .

Parameter	Type	Description
OS-DCF:diskConfig	String	Specifies the disk configuration. The value can be: • MANUAL: The BMS is created using the partitioning scheme and file system in the image. If the target flavor has a large disk, the remaining disk space cannot be partitioned. • AUTO: The BMS is created using a single partition with the same size as the disk of the target flavor. The file system can automatically adapt to the entire partition.
id	String	Specifies the BMS ID.
links	Array of objects	Specifies shortcut links of the BMS. For details, see Table 8-9 .
adminPass	String	Specifies the initial login password of the BMS administrator account.

Table 8-8 security_groups data structure

Parameter	Type	Description
name	String	Specifies the name of the security group that the BMS belongs to.

Table 8-9 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Example Response

```
{  
  "server": {
```

```
    "security_groups": [
      {
        "name": "default"
      }
    ],
    "OS-DCF:diskConfig": "MANUAL",
    "links": [
      {
        "rel": "self",
        "href": "https://openstack.example.com/v2/c685484a8cc2416b97260938705deb65/servers/9ab74d89-61e7-4259-8546-465fdebe4944"
      },
      {
        "rel": "bookmark",
        "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/servers/9ab74d89-61e7-4259-8546-465fdebe4944"
      }
    ],
    "id": "9ab74d89-61e7-4259-8546-465fdebe4944",
    "adminPass": "RjdD3h8U2DBe"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.3.2 Deleting a BMS (Deprecated)

Function

This API is used to delete a BMS.

URI

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

[Table 8-10](#) lists the parameters.

Table 8-10 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID.

Parameter	Mandatory	Description
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Deleting a BMS (ID: 9ab74d89-61e7-4259-8546-465fdebe4944)
DELETE https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/
9ab74d89-61e7-4259-8546-465fdebe4944

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.3.3 Querying Details About a BMS (Deprecated)

Function

This API is used to query details about a BMS based on the BMS ID.

This API has been deprecated. Use the API in [Querying Details About a BMS](#).

URI

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

Table 8-11 lists the parameters.

Table 8-11 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying details about a BMS (ID: 9ab74d89-61e7-4259-8546-465fdebe4944)
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/
9ab74d89-61e7-4259-8546-465fdebe4944

Response Parameters

Parameter	Type	Description
server	Object	Specifies BMS information. For details, see Table 8-12 .

Table 8-12 server data structure

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

Parameter	Type	Description
status	String	Specifies the BMS status. Value range: ● ACTIVE: Running, Stopping, Deleting ● BUILD: Creating ● ERROR: Faulty ● HARD_REBOOT: Forcibly Restarting ● REBOOT: Restarting ● SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen
created	String	Specifies the time when the BMS was created. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T03:30:52Z.
updated	String	Specifies the time when the BMS was last updated. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T04:30:52Z.
flavor	Object	Specifies the BMS flavor. For details, see Table 8-13 .
image	Object	Specifies the BMS image. For details, see Table 8-14 .
tenant_id	String	Specifies the ID of the tenant owning the BMS. The value is in UUID format. This parameter specifies the same meaning as project_id .
key_name	String	Specifies the SSH key name.
user_id	String	Specifies the ID of the user owning the BMS.
metadata	Map<String,String>	Specifies the BMS metadata. It is a user-defined metadata key-value pair. Each key or value contains a maximum of 255 characters.
hostId	String	Specifies the host ID of the BMS.

Parameter	Type	Description
addresses	Map<String,Array of address objects>	Specifies BMS network addresses. Specifies the VPC used by the BMS. • key: indicates the ID of the VPC used by the BMS. • value indicates the VPC details.
security_groups	Array of objects	Specifies security groups of the BMS. For details, see Table 8-17 .
links	Array of objects	Specifies shortcut links of the BMS. For details, see Table 8-15 .
OS-DCF:diskConfig	String	Specifies the disk configuration method. This is an extended attribute. The value can be: • MANUAL: The BMS is created using the partitioning scheme and file system in the image. If the target flavor has a large disk, the remaining disk space cannot be partitioned. • AUTO: The BMS is created using a single partition with the same size as the disk of the target flavor. The file system can automatically adapt to the entire partition.
OS-EXT-AZ:availability_zone	String	Specifies the AZ where the BMS is located.
OS-EXT-SRV-ATTR:host	String	Specifies the hostname of the BMS. This is an extended attribute.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the hostname of the hypervisor. This is an extended attribute provided by the Nova virt driver.
OS-EXT-SRV-ATTR:instance_name	String	Specifies the BMS alias. This is an extended attribute.
OS-EXT-STS:power_state	Integer	Specifies the BMS power status. This is an extended attribute. Options: 0, 1, 2, 3, and 4 • 0: pending • 1: running • 2: paused • 3: shutdown • 4: crashed

Parameter	Type	Description
OS-EXT-STS:task_state	String	Specifies the BMS task status. This is an extended attribute. Value range: • rebooting: The BMS is being restarted. • reboot_started: The BMS is normally restarted. • reboot_started_hard: The BMS is forcibly restarted. • powering-off: The BMS is being powered off. • powering-on: The BMS is being powered on. • rebuilding: The BMS is being rebuilt. • scheduling: The BMS is being scheduled. • deleting: The BMS is being deleted.
OS-EXT-STS:vm_state	String	Specifies the BMS status. This is an extended attribute. Value range: • active: The BMS is running. • stopped: The BMS is stopped.
OS-SRV-USG:launched_at	String	Specifies the time when the BMS was started. This is an extended attribute. The timestamp format is ISO 8601, for example, 2019-05-22T03:23:59.000000.
OS-SRV-USG:terminated_at	String	Specifies the time when the BMS was deleted. This is an extended attribute. The timestamp format is ISO 8601, for example, 2019-05-22T04:23:59.000000.
os-extended-volumes:volumes_attached	Array of objects	Specifies the EVS disks attached to the BMS. For details, see Table 8-18.
accessIPv4	String	This is a reserved attribute.
accessIPv6	String	This is a reserved attribute.
fault	Object	Specifies the fault cause. If the BMS is faulty, this field is returned. For details, see Table 8-19.

Parameter	Type	Description
config_drive	String	This is a reserved attribute.
progress	Integer	This is a reserved attribute.
description	String	Provides supplementary information about the pool. This parameter is added in micro version 2.19.
host_status	String	Specifies the status of the host that accommodates the BMS. • UP: The nova-compute is normal. • UNKNOWN: The nova-compute status is unknown. • DOWN: the nova-compute status is abnormal. • MAINTENANCE: The nova-compute is in the maintenance state. • Empty string: The BMS does not have host information. Added in micro version 2.16.
OS-EXT-SRV-ATTR:hostname	String	Specifies the hostname of the BMS. Added in micro version 2.3.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the reserved BMS IDs in a batch BMS creation situation. Added in micro version 2.3.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the BMS startup sequence in a batch BMS creation situation. Added in micro version 2.3.
OS-EXT-SRV-ATTR:kernel_id	String	Specifies the UUID of the kernel image when the AMI image is used. In other scenarios, leave this parameter blank. Added in micro version 2.3.
OS-EXT-SRV-ATTR:ramdisk_id	String	Specifies the UUID of the Ramdisk image when the AMI image is used. In other scenarios, leave this parameter blank. Added in micro version 2.3.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the BMS system disk, for example, /dev/sda . Added in micro version 2.3.

Parameter	Type	Description
OS-EXT-SRV-ATTR:user_data	String	Specifies the user_data specified during BMS creation. The value is encoded using Base64 or is an empty string.
locked	Boolean	Specifies whether the BMS is locked. • true: The BMS is locked. • false: The BMS is not locked. Added in micro version 2.9.
tags	Array of strings	Specifies tags of the BMS. This parameter is added in microversion 2.26. If the microversion is not used for query, the response does not contain the tags field. The value of this field meets the following requirements: • 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.

Table 8-13 flavor data structure

Parameter	Type	Description
id	String	Specifies the BMS type ID.
links	Array of objects	Specifies shortcut links of the BMS type. For details, see Table 8-15 .

Table 8-14 image data structure

Parameter	Type	Description
id	String	Specifies the image ID of the BMS.
links	Array of objects	Specifies shortcut links of the BMS image. For details, see Table 8-15 .

Table 8-15 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Table 8-16 address data structure

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	Specifies the type of the IP address. The value can be 4 or 6 . • 4: The type of the IP address is IPv4. • 6: The type of the IP address is IPv6.
OS-EXT-IPS-MAC:mac_addr	String	Specifies the MAC address. This is an extended attribute.
OS-EXT-IPS:type	String	Specifies the IP address type. This is an extended attribute. • fixed: indicates the private IP address. • floating: indicates the EIP.

Table 8-17 security_groups data structure

Parameter	Type	Description
name	String	• If no security group is specified during BMS creation, the default value is used. • If a security group is specified when you create the BMS, the value of this parameter is the security group name.

Table 8-18 os-extended-volumes:volumes_attached data structure

Parameter	Type	Description
id	String	Specifies the EVS disk ID.
delete_on_termination	Boolean	Specifies whether to delete the disk when deleting the BMS. • true: Yes • false: No Added in micro version 2.3.

Table 8-19 fault data structure

Parameter	Type	Description
message	String	Specifies the fault information.
code	Integer	Specifies the fault code.
details	String	Specifies the fault details.
created	String	Specifies the time when the fault occurred. The time is in ISO 8601 format.

Example Response

```
{
  "server": {
    "tenant_id": "c685484a8cc2416b97260938705deb65",
    "addresses": {
      "08a7715f-7de6-4ff9-a343-95ba4209f24a": [
        {
          "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:0e:c3:77",
          "OS-EXT-IPS:type": "fixed",
          "addr": "192.168.0.107",
          "version": 4
        }
      ]
    },
    "metadata": {
      "op_svc_userid": "1311c433dd9b408886f57d695c229cbe"
    },
    "OS-EXT-STS:task_state": null,
    "OS-DCF:diskConfig": "MANUAL",
    "OS-EXT-AZ:availability_zone": "az-dc-1",
    "links": [
      {
        "rel": "self",
        "href": "https://openstack.example.com/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
      },
      {
        "rel": "bookmark",
        "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
      }
    ]
  }
}
```

```
],
"OS-EXT-STS:power_state": 1,
"id": "95bf2490-5428-432c-ad9b-5e3406f869dd",
"os-extended-volumes:volumes_attached": [
{
    "id": "dfa375b5-9856-44ad-a937-a4802b6434c3"
},
{
    "id": "bb9f1b27-843b-4561-b62e-ca18eeaec417"
},
{
    "id": "86e801c3-acc6-465d-890c-d43ba493f553"
},
{
    "id": "0994d3ac-3c6a-495c-a439-c597a4f08fa6"
}
],
"OS-EXT-SRV-ATTR:host": "bms.az1",
"image": {
    "links": [
{
    "rel": "bookmark",
    "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/images/1a6635d8-afea-4f2b-abb6-27a202bad319"
}
],
    "id": "1a6635d8-afea-4f2b-abb6-27a202bad319"
},
"OS-SRV-USG:terminated_at": null,
"accessIPv4": "",
"accessIPv6": "",
"created": "2017-05-24T06:14:05Z",
"hostId": "e9c3ee0fcc58ab6085cf30df70b5544eab958858fb50d925f023e53e",
"OS-EXT-SRV-ATTR:hypervisor_hostname": "nova004@2",
"key_name": "$key_name",
"flavor": {
    "links": [
{
    "rel": "bookmark",
    "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/flavors/physical.83.medium"
}
],
    "id": "physical.83.medium"
},
"security_groups": [
{
    "name": "0011b620-4982-42e4-ad12-47c95ca495c4"
}
],
"config_drive": "",
"OS-EXT-STS:vm_state": "active",
"OS-EXT-SRV-ATTR:instance_name": "instance-0000ebd3",
"user_id": "1311c433dd9b408886f57d695c229cbe",
"name": "bms-83",
"progress": 0,
"OS-SRV-USG:launched_at": "2017-05-25T03:40:25.066078",
"updated": "2017-05-25T03:40:25Z",
"status": "ACTIVE"
}
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.3.4 Querying BMSs (Deprecated)

Function

This API is used to query BMSs.

This API has been deprecated. Use the API in [Querying Details About BMSs](#).

Constraints

- The query result returned by this API includes both ECSs and BMSs. You need to filter out the BMSs using the flavor used to create the BMSs or the tags added to the BMSs during BMS creation.
- If the image is used as the search criteria, other search criteria and pagination 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.

URI

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

[Table 8-20](#) lists the parameters.

Table 8-20 Parameter description

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

Request Parameters

Parameter	Mandatory	Type	Description
changes-since	No	String	Specifies the timestamp of the last BMS status update. The parameter is in ISO 8601 time format, for example, 2013-06-09T06:42:18Z .
image	No	String	Specifies the image ID. You can obtain the image ID from the IMS console or by calling the Querying Images API. NOTE If the image is used as the search criteria, other search criteria and pagination 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 flavor ID. You can obtain the flavor ID from the BMS console or using the Querying BMS Flavors (Deprecated) API.
name	No	String	Specifies the BMS name. This parameter supports fuzzy matching. For example, the regular expression ?name=bob will return both bob and bobbb. To obtain only bob, you can use a regular expression matching the basic database syntax, such as MySQL or PostgreSQL (official website: https://www.postgresql.org/docs/9.2/static/functions-matching.html).

Parameter	Mandatory	Type	Description
status	No	String	Specifies the BMS status. Value range: • ACTIVE: Running, Stopping, Deleting • BUILD: Creating • ERROR: Faulty • HARD_REBOOT: Forcibly Restarting • REBOOT: Restarting • SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen
limit	No	Integer	Specifies the number of BMSs displayed on each page.
marker	No	String	Specifies the BMS ID to which the marker corresponds. The query will start from the next ID.
tags	No	String	Queries the BMSs with specified tags. Added in micro version 2.26.
not-tags	No	String	Queries the BMSs with tags not containing the specified value. The value is a list of tag keys. NOTE 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". Added in micro version 2.26.
reservation_id	No	String	Specifies the reserved ID, which can be used to query BMSs created in a batch. Added in micro version 2.26.

Parameter	Mandatory	Type	Description
sort_key	No	String	Specifies the BMS sorting attribute, which can be the BMS UUID (uuid), BMS status (vm_state), BMS name (display_name), BMS task status (task_state), power status (power_state), creation time (created_at), last time when the BMS is updated (updated_at), and AZ (availability_zone). You can specify multiple sort_key and sort_dir pairs. The default sorting is the reverse order by created_at .
sort_dir	No	String	Specifies the sorting direction. • asc: The query results are displayed in ascending order. • desc (default value): The query results are displayed in descending order.

Example Request

- Querying BMSs
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers
- Querying BMSs with the **__type_baremetal** tag
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers?
tags=__type_baremetal
- Querying BMSs with the **__type_baremetal** tag and with **bms-test01** contained in the name
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers?
tags=__type_baremetal&name=bms-test01

Response Parameters

Parameter	Type	Description
servers	Array of objects	Specifies the BMS list. For details, see Table 8-21 .

Table 8-21 servers data structure

Parameter	Type	Description
name	String	Specifies the BMS name.

Parameter	Type	Description
id	String	Specifies the unique ID of the BMS.
links	Array of objects	Specifies shortcut links of the BMS. For details, see Table 8-22 .

Table 8-22 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Example Response

```
{
  "servers": [
    {
      "name": "bms",
      "links": [
        {
          "rel": "self",
          "href": "https://openstack.example.com/v2.1/c685484a8cc2416b97260938705deb65/servers/820abbd0-2d8b-4bc5-ae46-69cacfd4fbaa"
        },
        {
          "rel": "bookmark",
          "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/servers/820abbd0-2d8e-4bc5-ae46-69cacfd4fbaa"
        }
      ],
      "id": "820abbd0-2d8e-4bc5-ae46-69cacfd4fbaa"
    }
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.3.5 Querying Details About BMSs (Deprecated)

Function

This API is used to query details about BMSs.

This API has been deprecated. Use the API in [Querying Details About BMSs](#).

Constraints

- The query result returned by this API includes both ECSs and BMSs. You need to filter out the BMSs using the flavor used to create the BMSs or the tags added to the BMSs during BMS creation.
- If the image is used as the search criteria, other search criteria and pagination 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.

URI

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

[Table 8-23](#) lists the parameters.

Table 8-23 Parameter description

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

Request Parameters

Parameter	Mandatory	Type	Description
changes-since	No	String	Specifies the timestamp of the last BMS status update. The parameter is in ISO 8601 time format, for example, 2013-06-09T06:42:18Z .

Parameter	Mandatory	Type	Description
image	No	String	Specifies the image ID. You can obtain the image ID from the IMS console or by calling the Querying Images API. NOTE If the image is used as the search criteria, other search criteria and pagination 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 flavor ID. You can obtain the flavor ID from the BMS console or using the Querying BMS Flavors (Deprecated) API.
name	No	String	Specifies the BMS name. This parameter supports fuzzy matching. For example, the regular expression ?name=bob will return both bob and bobbb. To obtain only bob, you can use a regular expression matching the basic database syntax, such as MySQL or PostgreSQL (official website: https://www.postgresql.org/docs/9.2/static/functions-matching.html).

Parameter	Mandatory	Type	Description
status	No	String	Specifies the BMS status. Value range: • ACTIVE: Running, Stopping, Deleting • BUILD: Creating • ERROR: Faulty • HARD_REBOOT: Forcibly Restarting • REBOOT: Restarting • SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen
limit	No	Integer	Specifies the number of BMSs displayed on each page.
marker	No	String	Specifies the BMS ID to which the marker corresponds. The query will start from the next ID.
tags	No	String	Queries the BMSs with specified tags. Added in micro version 2.26.
not-tags	No	String	Queries the BMSs with tags not containing the specified value. The value is a list of tag keys. NOTE 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". Added in micro version 2.26.
reservation_id	No	String	Specifies the reserved ID, which can be used to query BMSs created in a batch. Added in micro version 2.26.

Parameter	Mandatory	Type	Description
sort_key	No	String	Specifies the BMS sorting attribute, which can be the BMS UUID (uuid), BMS status (vm_state), BMS name (display_name), BMS task status (task_state), power status (power_state), creation time (created_at), last time when the BMS is updated (updated_at), and AZ (availability_zone). You can specify multiple sort_key and sort_dir pairs. The default sorting is the reverse order by created_at .
sort_dir	No	String	Specifies the sorting direction. • asc: The query results are displayed in ascending order. • desc (default value): The query results are displayed in descending order.

Example Request

- Querying details about BMSs
`https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/detail`
- Querying details about BMSs with the **__type_baremetal** tag
`https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/detail?tags=__type_baremetal`
- Querying details about BMSs with the **__type_baremetal** tag and with **bms-test01** contained in the name
`https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/detail?tags=__type_baremetal&name=bms-test01`

Response Parameters

Parameter	Type	Description
servers	Array of objects	Specifies details about the BMS. For details, see Table 8-24 .

Table 8-24 server data structure

Parameter	Type	Description
name	String	Specifies the BMS name.
id	String	Specifies the unique ID of the BMS.
status	String	Specifies the BMS status. Value range: • ACTIVE: The BMS is running, stopped, or being deleted. • BUILD: The BMS is being created. • ERROR: The BMS is faulty. • HARD_REBOOT: The BMS is being forcibly restarted. • REBOOT: The BMS is being restarted. • SHUTOFF: The BMS is stopped, being started, being deleted, being rebuilt, being reinstalled, failed to be reinstalled, or frozen.
created	String	Specifies when the BMS was created. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T03:30:52Z.
updated	String	Specifies the last time when the BMS was updated. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T04:30:52Z.
flavor	Object	Specifies the BMS flavor. For details, see Table 8-27 .
image	Object	Specifies the BMS image. For details, see Table 8-28 .
tenant_id	String	Specifies the ID of the tenant owning the BMS. The value is in UUID format. This parameter equals to project_id .
key_name	String	Specifies the SSH key name.
user_id	String	Specifies the ID of the user owning the BMS.
metadata	Map<String,String>	Specifies the BMS metadata. It is a user-defined metadata key-value pair. Each key or value contains a maximum of 255 characters.

Parameter	Type	Description
hostId	String	Specifies the host ID of the BMS.
addresses	Map<String,Array of address objects>	Specifies the network address of the BMS. Specifies the VPC used by the BMS. • key: indicates the ID of the VPC used by the BMS. • value indicates the VPC details.
security_groups	Array of objects	Specifies security groups of the BMS. For details, see Table 8-31 .
links	Array of objects	Specifies shortcut links of the BMS. For details, see Table 8-29 .
OS-DCF:diskConfig	String	Specifies the disk configuration method. This is an extended attribute. The value can be: • MANUAL: The BMS is created using the partitioning scheme and file system in the image. If the target flavor has a large disk, the API does not partition the remaining disk space. • AUTO: The BMS is created using a single partition with the same size as the disk of the target flavor. The API automatically adjusts the file system to adapt to the entire partition.
OS-EXT-AZ:availability_zone	String	Specifies the name of the AZ where the BMS is deployed. This is an extended attribute.
OS-EXT-SRV-ATTR:host	String	Specifies the hostname of the BMS. This is an extended attribute.
OS-EXT-SRV-ATTR:hypervisor_hostname	String	Specifies the hostname of the hypervisor. This is an extended attribute provided by the Nova virt driver.
OS-EXT-SRV-ATTR:instance_name	String	Specifies the BMS alias. This is an extended attribute.

Parameter	Type	Description
OS-EXT-STS:power_state	Integer	Specifies the BMS power status. This is an extended attribute. Value range: • 0: pending • 1: running • 2: paused • 3: shutdown • 4: crashed
OS-EXT-STS:task_state	String	Specifies the BMS task status. This is an extended attribute. Value range: • rebooting: The BMS is being restarted. • reboot_started: The BMS is normally restarted. • reboot_started_hard: The BMS is forcibly restarted. • powering-off: The BMS is being powered off. • powering-on: The BMS is being powered on. • rebuilding: The BMS is being rebuilt. • scheduling: The BMS is being scheduled. • deleting: The BMS is being deleted.
OS-EXT-STS:vm_state	String	Specifies the BMS status. This is an extended attribute. Value range: • active: The BMS is running. • shutoff: The BMS is stopped. • reboot: The BMS is restarted.
OS-SRV-USG:launched_at	String	Specifies the time when the BMS was started. This is an extended attribute. The timestamp format complies with ISO 8601, for example, 2019-05-22T03:23:59.000000 .
OS-SRV-USG:terminated_at	String	Specifies the time when the BMS was deleted. This is an extended attribute. The timestamp format complies with ISO 8601, for example, 2019-05-22T04:23:59.000000 .

Parameter	Type	Description
os-extended-volumes:volumes_attached	Array of objects	Specifies the attached EVS disks of the BMS. For details, see Table 8-32 .
accessIPv4	String	This is a reserved attribute.
accessIPv6	String	This is a reserved attribute.
fault	Object	Specifies the fault cause. If the BMS is faulty, this field will be returned. For details, see Table 8-33 .
config_drive	String	This is a reserved attribute.
progress	Integer	This is a reserved attribute.
description	String	Specifies description about the BMS. This is a new parameter in micro version 2.19.
host_status	String	Specifies the status of the host that accommodates the BMS. • UP: The host is normal. • UNKNOWN: The host status is unknown. • DOWN: the host is abnormal. • MAINTENANCE: The host is being maintained. • Empty string: There is no BMS host information. This is a new parameter in micro version 2.16.
OS-EXT-SRV-ATTR:hostname	String	Specifies the hostname of the BMS. This is a new parameter in micro version 2.3.
OS-EXT-SRV-ATTR:reservation_id	String	Specifies the reserved BMS IDs when BMSs are created in a batch. This is a new parameter in micro version 2.3.
OS-EXT-SRV-ATTR:launch_index	Integer	Specifies the startup sequence of the BMSs created in a batch. This is a new parameter in micro version 2.3.

Parameter	Type	Description
OS-EXT-SRV-ATTR:kernel_id	String	Specifies the UUID of the kernel image when an AMI image is used. In other cases, leave this parameter blank. This is a new parameter in micro version 2.3.
OS-EXT-SRV-ATTR:ramdisk_id	String	Specifies the UUID of a RAM disk image when the AMI image is used. In other cases, leave this parameter blank. This is a new parameter in micro version 2.3.
OS-EXT-SRV-ATTR:root_device_name	String	Specifies the device name of the BMS system disk, for example, /dev/sda . This is a new parameter in micro version 2.3.
OS-EXT-SRV-ATTR:user_data	String	Specifies the user_data specified during BMS creation. The value is encoded using Base64 or an empty string.
locked	Boolean	Specifies whether the BMS is locked. • true: The BMS is locked. • false: The BMS is not locked. This is a new parameter in micro version 2.9.
tags	Array of strings	Specifies tags of the BMS. This is a new parameter in micro version 2.26. If you are not using this micro version, the response will not contain the tags parameter. A tag must meet the following requirements: • 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.
sys_tags	Array of objects	Specifies system tags of the BMS. For details, see Table 8-25 .
enterprise_project_id	String	Specifies the enterprise project ID.
os:scheduler_hints	Object	Specifies scheduling information of the BMS. For details, see Table 8-26 .

Table 8-25 sys_tags data structure

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

Table 8-26 os:scheduler_hints data structure

Parameter	Type	Description
group	Array of strings	Specifies the BMS group ID in UUID format.

Table 8-27 flavor data structure

Parameter	Type	Description
id	String	Specifies the BMS type ID.
links	Array of objects	Specifies shortcut links of the BMS type. For details, see Table 8-29 .

Table 8-28 image data structure

Parameter	Type	Description
id	String	Specifies the image ID of the BMS.
links	Array of objects	Specifies shortcut links of the BMS image. For details, see Table 8-29 .

Table 8-29 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.

Parameter	Type	Description
href	String	Specifies the corresponding shortcut link.

Table 8-30 address data structure

Parameter	Type	Description
addr	String	Specifies the IP address.
version	Integer	Specifies the type of the IP address. The value can be 4 or 6 . • 4: The type of the IP address is IPv4. • 6: The type of the IP address is IPv6.
OS-EXT-IPS-MAC:mac_addr	String	Specifies the MAC address. This is an extended attribute.
OS-EXT-IPS:type	String	Specifies the IP address type. This is an extended attribute. • fixed: indicates the private IP address. • floating: indicates the EIP.

Table 8-31 security_groups data structure

Parameter	Type	Description
name	String	• If no security group is specified during BMS creation, the default value is used. • If a security group is specified when you create the BMS, the value of this parameter is the security group name.

Table 8-32 os-extended-volumes:volumes_attached data structure

Parameter	Type	Description
id	String	Specifies the EVS disk ID.

Parameter	Type	Description
delete_on_termination	Boolean	Specifies whether to delete the disk when deleting the BMS. • true: Yes • false: No Added in micro version 2.3.

Table 8-33 fault data structure

Parameter	Type	Description
message	String	Specifies the fault information.
code	Integer	Specifies the fault code.
details	String	Specifies the fault details.
created	String	Specifies the time when the fault occurred. The time is in ISO 8601 format.

Example Response

```
{
  "servers": [
    {
      "tenant_id": "c685484a8cc2416b97260938705deb64",
      "addresses": {
        "08a7715f-7de6-4ff9-a343-95ba4209f24a": [
          {
            "OS-EXT-IPS-MAC:mac_addr": "fa:16:3e:0e:c4:77",
            "OS-EXT-IPS:type": "fixed",
            "addr": "192.168.0.107",
            "version": 4
          }
        ]
      },
      "metadata": {
        "op_svc_userid": "1311c433dd9b408886f57d695c229cbe"
      },
      "OS-EXT-STS:task_state": null,
      "OS-DCF:diskConfig": "MANUAL",
      "OS-EXT-AZ:availability_zone": "az-dc-1",
      "links": [
        {
          "rel": "self",
          "href": "https://openstack.example.com/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
        },
        {
          "rel": "bookmark",
          "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
        }
      ],
      "OS-EXT-STS:power_state": 1,
      "id": "95bf2490-5428-432c-ad9b-5e3406f869dd",
    }
  ]
}
```

```

    "os-extended-volumes:volumes_attached": [
    {
        "id": "dfa375b5-9856-44ad-a937-a4802b6434c3"
    },
    {
        "id": "bb9f1b27-843b-4561-b62e-ca18eeaec417"
    },
    {
        "id": "86e801c3-acc6-465d-890c-d43ba493f553"
    },
    {
        "id": "0994d3ac-3c6a-495c-a439-c597a4f08fa6"
    }
    ],
    "OS-EXT-SRV-ATTR:host": "bms.az1",
    "image": {
        "links": [
        {
            "rel": "bookmark",
            "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/images/1a6635d8-afea-4f2b-abb6-27a202bad319"
        },
        {
            "id": "1a6635d8-afea-4f2b-abb6-27a202bad319"
        }
        ],
        "OS-SRV-USG:terminated_at": null,
        "accessIPv4": "",
        "accessIPv6": "",
        "created": "2017-05-24T06:14:05Z",
        "hostId": "e9c3ee0fcc58ab6085cf30df70b5544eab958858fb50d925f023e53e",
        "OS-EXT-SRV-ATTR:hypervisor_hostname": "nova004@2",
        "key_name": "$key_name",
        "flavor": {
            "links": [
            {
                "rel": "bookmark",
                "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/flavors/physical.83.medium"
            },
            {
                "id": "physical.83.medium"
            }
            ],
            "security_groups": [
            {
                "name": "0011b620-4982-42e4-ad12-47c95ca495c4"
            }
            ],
            "config_drive": "",
            "OS-EXT-STS:vm_state": "active",
            "OS-EXT-SRV-ATTR:instance_name": "instance-0000ebd3",
            "user_id": "1311c433dd9b408886f57d695c229cbe",
            "name": "bms",
            "progress": 0,
            "OS-SRV-USG:launched_at": "2017-05-25T03:40:25.066078",
            "updated": "2017-05-25T03:40:25Z",
            "status": "ACTIVE"
        }
    }
    ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.4 BMS Status Management (OpenStack Nova APIs)

8.4.1 Changing the Name of a BMS (Deprecated)

Function

This API is used to modify BMS information. Currently, only the name of a BMS can be changed.

This API has been deprecated. Use the API in [Changing a BMS Name](#).

URI

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

[Table 8-34](#) lists the parameters.

Table 8-34 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID.
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
server	Yes	Object	Specifies the BMS data structure. For details, see Table 8-35 .

Table 8-35 server data structure

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the new BMS name.

Example Request

Changing the name of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd) to **new-server-test**

```
PUT https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd
{
  "server": {
    "name": "new-server-test"
  }
}
```

Response Parameters

Table 8-36 Response parameters

Parameter	Type	Description
server	Object	Specifies BMS information. For details, see Table 8-37 .

Table 8-37 server data structure

Parameter	Type	Description
name	String	Specifies the BMS name.
id	String	Specifies the unique ID of the BMS.
status	String	Specifies the BMS status. • ACTIVE: Running, Stopping, Deleting • BUILD: Creating • ERROR: Faulty • HARD_REBOOT: Forcibly Restarting • REBOOT: Restarting • SHUTOFF: Stopped, Starting, Deleting, Rebuilding, Reinstalling OS, OS Reinstallation Failed, Frozen

Parameter	Type	Description
created	String	Specifies the time when the BMS was created. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T03:30:52Z.
updated	String	Specifies the last time when the BMS was updated. The timestamp format is YYYY-MM-DDTHH:MM:SSZ (ISO 8601), for example, 2019-05-22T04:30:52Z.
flavor	Object	Specifies the BMS flavor. For details, see Table 8-38 .
image	Object	Specifies the BMS image. For details, see Table 8-39 .
tenant_id	String	Specifies the ID of the tenant owning the BMS. The value is in UUID format. This parameter specifies the same meaning as project_id .
user_id	String	Specifies the ID of the user owning the BMS.
metadata	Object	Specifies the BMS metadata. For details, see Table 8-40 .
hostId	String	Specifies the host ID of the BMS.
addresses	Array of address objects	Specifies the VPC used by the BMS. • key: indicates the ID of the VPC used by the BMS. • value: indicates the VPC details. For details, see Table 8-42.
links	Array of objects	Specifies shortcut links of the BMS. For details, see Table 8-41 .
accessIPv4	String	This is a reserved attribute.
accessIPv6	String	This is a reserved attribute.

Parameter	Type	Description
OS-DCF:diskConfig	String	Specifies the disk configuration method. This is an extended attribute. The value can be: • MANUAL: The BMS is created using the partitioning scheme and file system in the image. If the target flavor has a large disk, the remaining disk space cannot be partitioned. • AUTO: The BMS is created using a single partition with the same size as the disk of the target flavor. The file system can automatically adapt to the entire partition.
progress	Integer	This is a reserved attribute.

Table 8-38 flavor data structure

Parameter	Type	Description
id	String	Specifies the flavor ID.
links	Array of objects	Specifies the shortcut link of the BMS flavor. For details, see Table 8-41 .

Table 8-39 image data structure

Parameter	Type	Description
id	String	Specifies the image ID of the BMS.
links	Array of objects	Specifies shortcut links of the BMS image. For details, see Table 8-41 .

Table 8-40 metadata data structure

Parameter	Type	Description
User-defined field key and value pair	String	Specifies the key and value pair of the metadata. Each key or value contains a maximum of 255 characters.

Table 8-41 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. The value can be: • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Table 8-42 address data structure

Parameter	Type	Description
addr	String	Specifies the IP address.
version	String	Specifies the version of the IP address.

Example Response

```
{
  "server": {
    "tenant_id": "c685484a8cc2416b97260938705deb65",
    "image": {
      "links": [
        {
          "rel": "bookmark",
          "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/images/1a6635d8-afea-4f2b-abb6-27a202bad319"
        }
      ],
      "id": "1a6635d8-afea-4f2b-abb6-27a202bad319"
    },
    "accessIPv4": "",
    "addresses": {
      "08a7715f-7de6-4ff9-a343-95ba4209f24a": [
        {
          "addr": "192.168.0.107",
          "version": 4
        }
      ]
    },
    "metadata": {
    },
    "accessIPv6": "",
    "created": "2017-05-24T06:14:05Z",
    "hostId": "e9c3ee0fcc58ab6085cf30df70b5544eab958858fb50d925f023e53e",
    "flavor": {
      "links": [
        {
          "rel": "bookmark",
```

```
        "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/flavors/physical.83.medium"
      },
      "id": "physical.83.medium"
    },
    "OS-DCF:diskConfig": "MANUAL",
    "user_id": "1311c433dd9b408886f57d695c229cbe",
    "name": "new-server-test",
    "progress": 0,
    "links": [
      {
        "rel": "self",
        "href": "https://openstack.example.com/v2/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
      },
      {
        "rel": "bookmark",
        "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd"
      }
    ],
    "id": "95bf2490-5428-432c-ad9b-5e3406f869dd",
    "updated": "2017-05-25T03:40:25Z",
    "status": "ACTIVE"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.4.2 Starting a BMS (Deprecated)

Function

This API is used to start a single BMS.

This API has been deprecated. Use the API in [Starting BMSs](#).

URI

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

[Table 8-43](#) lists the parameters.

Table 8-43 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
os-start	Yes	null	Specifies the operation of starting the BMS. The data structure is empty.

Example Request

```
Starting a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
POST https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/action
{
  "os-start": {}
}
```

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.4.3 Restarting a BMS (Deprecated)

Function

This API is used to restart a single BMS.

This API has been deprecated. Use the API in [Restarting BMSs](#).

Constraints

Currently, only forcible restart is supported.

URI

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

[Table 8-44](#) lists the parameters.

Table 8-44 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
reboot	Yes	Object	Specifies the operation of restarting the BMS. For details, see Table 8-45 .

Table 8-45 reboot data structure

Parameter	Mandator y	Type	Description
type	Yes	String	Specifies the type of the restart operation. • SOFT: soft restart • HARD: forcible restart NOTE Currently, value SOFT is invalid. All BMS restart operations are forcible restart.

Example Request

```
Restarting a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
POST https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/action
{
  "reboot": {
    "type": "HARD"
  }
}
```

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.4.4 Stopping a BMS (Deprecated)

Function

This API is used to stop a single BMS. To stop multiple BMSs, you can use the [Stopping BMSs](#) API.

This API has been deprecated. Use the API in [Stopping BMSs](#).

Constraints

- The **OS-EXT-STS:vm_state** attribute (BMS status) must be **active** or **error**.
- Currently, only forcible stopping is supported.

URI

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

[Table 8-46](#) lists the parameters.

Table 8-46 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
os-stop	Yes	Object	Specifies the operation of stopping the BMS. For details, see Table 8-47 .

Table 8-47 os-stop data structure

Parameter	Mandator y	Type	Description
type	No	String	Specifies the type of the BMS stopping operation. • SOFT: normal BMS stopping • HARD: Forcible BMS stopping NOTE Currently, this parameter is invalid. All BMS stopping operations are forcible stopping.

Example Request

Stopping a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
POST https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/action
{
 "os-stop": {}
}

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.4.5 Obtaining a Remote Login Address (Deprecated)

Function

This API is used to obtain the address for remotely logging in to a BMS.

URL

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

[Table 8-48](#) lists the parameters.

Table 8-48 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Message

- Request parameters

Parameter	Mandatory	Type	Description
os-getSerialConsole	Yes	Object	Specifies the action to obtain a BMS remote login address. For details, see Table 8-49 .

Table 8-49 os-getSerialConsole data structure

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the object. Set the value to serial .

- Example request

Obtaining the remote login address of a BMS (ID: 47e9be4e-a7b9-471f-92d9-ffc83814e07a)

POST https://{ECS Endpoint}/v2/9c53a566cb3443ab910cf0daebca90c4/servers/47e9be4e-a7b9-471f-92d9-ffc83814e07a/action

```
{
  "os-getSerialConsole": {
    "type": "serial"
  }
}
```

Response Message

- Response parameters

None

- Example response

```
{
  "console": {
    "url": "https://baremetal-consoleproxy.az1.dc1.domainname.com:8003/?token=040134bb-9195-4029-9a62-550bce390258",
    "type": "serial"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.5 BMS Metadata Management (OpenStack Nova APIs)

8.5.1 Querying BMS Metadata (Deprecated)

Function

The BMS metadata includes BMS basic information on the cloud platform, such as the BMS ID, hostname, and network information. This API is used to query metadata of a BMS.

This API has been deprecated. Use the API in [Querying Details About a BMS](#).

Constraints

Pagination query is not supported.

URI

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

[Table 8-50](#) lists the parameters.

Table 8-50 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying metadata of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/metadata

Response Parameters

Parameter	Type	Description
metadata	Map<String,String>	Specifies a user-defined metadata key-value pair. Each key or value contains a maximum of 255 characters.

Example Response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.5.2 Updating BMS Metadata (Deprecated)

Function

This API is used to update BMS metadata.

- If the metadata does not contain the target field, the field is automatically added to the field.
- If the metadata contains the target field, the field value is automatically updated.

This API has been deprecated. Use the API in [Updating BMS Metadata](#).

Constraints

The BMS **OS-EXT-STS:vm_state** attribute (BMS status) must be **active**, **stopped**, or **paused**.

URI

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

[Table 8-51](#) lists the parameters.

Table 8-51 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
metadata	Yes	Object	Specifies the user-defined metadata key and value pair. For details, see Table 8-52 .

Table 8-52 metadata data structure

Parameter	Mandatory	Type	Description
User-defined field key and value pair	Yes	String	Specifies the user-defined metadata key and value pair. - Each key or value contains a maximum of 255 characters. - The key does not support the following special characters: :~!@#\$%^&*()=+<, >?/";{} \ - The value does not support the following special characters: \"

Example Request

Updating the metadata of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd) to customize a key-value pair

POST https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/metadata

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

Response Parameters

Parameter	Type	Description
metadata	Object	Specifies the user-defined metadata key and value pair. For details, see Table 8-53 .

Table 8-53 metadata data structure

Parameter	Type	Description
User-defined field key and value pair	String	Specifies the key and value pair of the metadata. - Each key or value contains fewer than 256 characters and cannot be left blank. - The value does not support the following special characters: \"

Example Response

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

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.5.3 Modifying Specified BMS Metadata (Deprecated)

Function

This API is used to modify specified BMS metadata.

This API has been deprecated. Use the API in [Updating BMS Metadata](#).

Constraints

The BMS **OS-EXT-STS:vm_state** attribute (BMS status) must be **active**, **stopped**, or **paused**.

URI

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

Table 8-54 lists the parameters.

Table 8-54 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
key	Yes	Specifies the BMS metadata key value to be modified.

Request Parameters

Parameter	Mandatory	Type	Description
meta	Yes	Object	Specifies the user-defined metadata key and value pair. For details, see Table 8-55 .

Table 8-55 meta data structure

Parameter	Mandatory	Type	Description
User-defined field key and value pair	Yes	String	Specifies the user-defined metadata key and value pair. - Each key or value contains a maximum of 255 characters. - The key does not support the following special characters: :~!@#\$\$%^&*()=+<, >?/'"; {[]}\ - The value does not support the following special characters: \"

Example Request

Modifying the metadata of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd) to customize a key-value pair

```
PUT https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/metadata/{key}
{
  "meta": {
    "key": "value"
  }
}
```

Response Parameters

Parameter	Type	Description
meta	Object	Specifies the user-defined metadata key and value pair. For details, see Table 8-56 .

Table 8-56 meta data structure

Parameter	Type	Description
User-defined field key and value pair	String	Specifies the user-defined metadata key and value pair. Each key or value contains a maximum of 255 characters.

Example Response

```
{
  "meta": {
    "key": "value"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.5.4 Deleting Specified BMS Metadata (Deprecated)

Function

This API is used to delete specified BMS metadata.

Constraints

The BMS **OS-EXT-STS:vm_state** attribute (BMS status) must be **active**, **stopped**, or **paused**.

URI

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

[Table 8-57](#) lists the parameters.

Table 8-57 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
key	Yes	Specifies the BMS metadata key value to be deleted.

Request Parameters

None

Example Request

Deleting metadata of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
DELETE https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/metadata/{key}

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.6 BMS IP Address Query (OpenStack Nova APIs)

8.6.1 Querying IP Addresses of a BMS (Deprecated)

Function

This API is used to query private IP addresses of a BMS.

This API has been deprecated. Use the API in [Querying Details About a BMS](#).

Constraints

Pagination query is not supported.

URI

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

[Table 8-58](#) lists the parameters.

Table 8-58 Parameter description

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

Parameter	Mandatory	Description
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying the private IP addresses of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/ips

Response Parameters

Parameter	Type	Description
addresses	Map<String,Array of address objects>	Specifies the VPC used by the BMS. • key: indicates the ID of the VPC used by the BMS. • value indicates the VPC details.

Table 8-59 address parameter structure description

Parameter	Type	Description
version	Integer	Specifies the IP address version. The value can be: • 4: IPv4 address • 6: IPv6 address
addr	String	Specifies the IP address.

Example Response

```
{
  "addresses": {
    "08a7715f-7de6-4ff9-a343-95ba4209f24a": [
      {
        "version": 4,
        "addr": "192.168.2.90"
      }
    ]
  }
}
```

```
    ]
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.6.2 Querying the Specified IP Address of a BMS (Deprecated)

Function

This API is used to query the specified IP address of a BMS based on the network name.

This API has been deprecated. Use the API in [Querying Details About a BMS](#).

URI

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

[Table 8-60](#) lists the parameters.

Table 8-60 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Parameter	Mandatory	Description
vpc_id	Yes	Specifies the ID of the VPC where the BMS is located.

Request Parameters

None

Example Request

Querying the IP address of a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd) in a specified VPC

GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/ips/{vpc_id}

Response Parameters

Parameter	Type	Description
Vpc_id	Array of address objects	Specifies the VPC used by the BMS. Vpc_id indicates the ID of the VPC used by the BMS.

Table 8-61 Network parameter structure description

Parameter	Type	Description
version	Integer	Specifies the IP address version. The value can be: • 4: IPv4 address • 6: IPv6 address
addr	String	Specifies the IP address.

Example Response

```
{
  "Vpc_id": [
    {
      "version": 4,
      "addr": "192.168.1.159"
    }
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.7 BMS Flavor Query (OpenStack Nova APIs)

8.7.1 Querying BMS Flavors (Deprecated)

Function

This API is used to query BMS flavors.

This API has been deprecated. Use the API in [Querying Flavor Details and Extended Flavor Information](#).

Constraints

The flavors you obtained using this API are all the flavors in the system. The flavors whose names starting with **physical** are BMS flavors and can be used to create BMSs.

URI

GET /v2.1/{project_id}/flavors/detail?
minDisk={minDisk}&minRam={minRam}&sort_key={sort_key}&sort_dir={sort_dir}

[Table 8-62](#) lists the parameters.

Table 8-62 Parameter description

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

Request Parameters

[Table 8-63](#) lists the optional parameters that can be used to query BMS flavors.

Table 8-63 Optional parameters

Parameter	Mandatory	Type	Description
minDisk	No	String	Specifies the minimum disk size in GB. Only the BMSs with a disk size greater than or equal to the minimum size can be queried.
minRam	No	String	Specifies the minimum memory size in MB. Only the BMSs with the memory size greater than or equal to the minimum size can be queried.
sort_key	No	String	Specifies the sorting field. The default value is flavorid . The value of this parameter can also be name , memory_mb , vcpus , root_gb , or flavorid .
sort_dir	No	String	Specifies the sorting of BMS flavors. The value can be asc or desc , and is asc by default.

Example Request

- Querying BMS flavors
GET <https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/flavors/detail>
- Querying BMS flavors with the minimum disk of 3,725 GB
GET <https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/flavors/detail?minDisk=3725>

Response Parameters

Parameter	Type	Description
flavors	Array of objects	Specifies BMS flavors. For details, see Table 8-64 .

Table 8-64 flavors data structure

Parameter	Type	Description
id	String	Specifies the BMS flavor ID.
name	String	Specifies the BMS flavor name.

Parameter	Type	Description
vcpus	Integer	Specifies the number of CPU cores in the BMS flavor.
ram	Integer	Specifies the memory size (MB) in the BMS flavor.
disk	Integer	Specifies the disk size (GB) in the BMS flavor.
swap	String	This is a reserved attribute.
OS-FLV-EXT-DATA:ephemeral	Integer	This is a reserved attribute.
OS-FLV-DISABLED:disable d	Boolean	This is a reserved attribute.
rxtx_factor	Float	This is a reserved attribute.
os-flavor-access:is_public	Boolean	This is a reserved attribute.
links	Array of objects	Specifies shortcut links of the BMS flavor. For details, see Table 8-65 .

Table 8-65 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. • self: resource link that contains the version number. It is used when immediate tracing is required. • bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Example Response

```
{
  "flavors": [
    {
      "name": "physical.o2.medium",
      "links": [
        {
          "href": "https://openstack.example.com/v2/c685484a8cc2416b97260938705deb65/flavors/physical.o2.medium",
          "rel": "self"
        }
      ]
    }
  ]
}
```

```
        {
            "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/flavors/physical.o2.medium",
            "rel": "bookmark"
        }
    ],
    "ram": 321725,
    "OS-FLV-DISABLED:disabled": false,
    "vcpus": 56,
    "swap": "",
    "os-flavor-access:is_public": true,
    "rxtx_factor": 1,
    "OS-FLV-EXT-DATA:ephemeral": 0,
    "disk": 3725,
    "id": "physical.o2.medium"
}
]
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.7.2 Querying Details About a BMS Flavor (Deprecated)

Function

This API is used to query details about a BMS flavor, such as the flavor name, CPU cores, and memory.

This API has been deprecated. Use the API in [Querying Flavor Details and Extended Flavor Information](#).

URI

GET /v2.1/{project_id}/flavors/{flavor_id}

[Table 8-66](#) lists the parameters.

Table 8-66 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
flavor_id	Yes	Specifies the flavor ID. You can obtain the flavor ID from the BMS console or using the Querying BMS Flavors (Deprecated) API.

Request Parameters

None

Example Request

Query details about a flavor (ID: physical.o2.medium)
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/flavors/physical.o2.medium

Response Parameters

Parameter	Type	Description
flavor	Object	Specifies a BMS flavor. For details, see Table 8-67 .

Table 8-67 flavor field description

Parameter	Type	Description
id	String	Specifies the BMS flavor ID.
name	String	Specifies the BMS flavor name.
vcpus	Integer	Specifies the number of CPU cores in the BMS flavor.
ram	Integer	Specifies the memory size (MB) in the BMS flavor.
disk	Integer	Specifies the disk size (GB) in the BMS flavor.
swap	String	This is a reserved attribute.

Parameter	Type	Description
OS-FLV-EXT-DATA:ephemeral	Integer	This is a reserved attribute.
OS-FLV-DISABLED:disabled	Boolean	This is a reserved attribute.
rxtx_factor	Float	This is a reserved attribute.
os-flavor-access:is_public	Boolean	This is a reserved attribute.
links	Array of objects	Specifies shortcut links of the BMS flavor. For details, see Table 8-68 .

Table 8-68 links data structure

Parameter	Type	Description
rel	String	Specifies the shortcut link marker name. ● self: resource link that contains the version number. It is used when immediate tracing is required. ● bookmark: resource link that can be stored for a long time.
href	String	Specifies the corresponding shortcut link.

Example Response

```
{
  "flavor": {
    "name": "physical.o2.medium",
    "links": [
      {
        "href": "https://openstack.example.com/v2/c685484a8cc2416b97260938705deb65/flavors/physical.o2.medium",
        "rel": "self"
      },
      {
        "href": "https://openstack.example.com/c685484a8cc2416b97260938705deb65/flavors/physical.o2.medium",
        "rel": "bookmark"
      }
    ],
    "ram": 192705,
    "OS-FLV-DISABLED:disabled": false,
    "vcpus": 24,
    "swap": "",
    "os-flavor-access:is_public": true,
    "rxtx_factor": 1,
    "OS-FLV-EXT-DATA:ephemeral": 0,
  }
}
```

```
"disk": 1862,  
"id": "physical.o2.medium"  
}  
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.7.3 Querying Details About extra_specs Parameters of a BMS Flavor (Deprecated)

Function

extra_specs parameters are key-value pairs of a BMS flavor. For example, **baremetal:extBootType** indicates the boot device of a BMS. Its value can be **LocalDisk** (local disk) or **Volume** (EVS disk). If you want to check whether a flavor supports quick provisioning, you can call this API.

This API has been deprecated. Use the API in [Querying Flavor Details and Extended Flavor Information](#).

URI

GET /v2.1/{project_id}/flavors/{flavor_id}/os-extra_specs

[Table 8-69](#) lists the parameters.

Table 8-69 Parameter description

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

Parameter	Mandatory	Description
flavor_id	Yes	Specifies the flavor ID. You can obtain the flavor ID from the BMS console or using the Querying BMS Flavors (Deprecated) API.

Request Parameters

None

Example Request

Querying details about extra_specs parameters of a flavor (ID: physical.s2.medium)

```
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/flavors/physical.s2.medium/os-extra_specs
```

Response Parameters

Parameter	Type	Description
extra_specs	Object	Specifies the key-value pair of a BMS flavor. • capabilities:cpu_arch: specifies the CPU architecture of the BMS. The value can be x86_64 (for x86 servers) or aarch64 (for ARM servers). • baremetal:disk_detail: specifies the disk description. • capabilities:hypervisor_type: specifies the hypervisor type. The value is fixed at ironic. • baremetal:__support_evs: specifies whether to support EVS disks. The value can be true or false. If the flavor does not contain this parameter, EVS disks are not supported either. • baremetal:extBootType: specifies the boot device of the BMS. The value can be LocalDisk (local disk) or Volume (EVS disk). • baremetal:net_num: specifies the number of NICs that can be attached to a BMS. • baremetal:netcard_detail: specifies description of the NIC. • baremetal:cpu_detail: specifies description of the CPU. • resource_type: specifies the resource type. The value is fixed at ironic. • baremetal:memory_detail: specifies description of the memory.

Example Response

```
{
  "extra_specs": {
    "capabilities:cpu_arch": "x86_64",
    "baremetal:disk_detail": "SAS 8T",
    "capabilities:hypervisor_type": "ironic",
    "baremetal:__support_evs": "true",
    "baremetal:extBootType": "LocalDisk",
    "capabilities:board_type": "s2m",
    "baremetal:net_num": "2",
    "baremetal:netcard_detail": "2*10GE",
    "baremetal:cpu_detail": "2*8coreIntel Xeon E5-2667 V43.2GHz",
    "resource_type": "ironic",
    "baremetal:memory_detail": "256GB DDR4 RAM(GB)"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.8 BMS NIC Management (OpenStack Nova APIs)

8.8.1 Querying Information About BMS NICs (Deprecated)

Function

This API is used to query information about BMS NICs, such as the MAC addresses and private IP addresses.

This API has been deprecated. Use the API in [Querying NICs Bound to a BMS](#).

URI

GET /v2.1/{project_id}/servers/{server_id}/os-interface

[Table 8-70](#) lists the parameters.

Table 8-70 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying information about NICs bound to a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)

```
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-interface
```

Response Parameters

Parameter	Type	Description
interfaceAttachments	Array of objects	Specifies information about NICs of the BMS. For details, see Table 8-71 .

Table 8-71 interfaceAttachments data structure

Parameter	Type	Description
port_state	String	Specifies the status of the NIC port. The value can be ACTIVE , BUILD , or DOWN .
fixed_ips	Array of objects	Specifies the NIC private IP address. For details, see Table 8-72 .
net_id	String	Specifies the ID of the subnet (network_id) to which the NIC ports belong.
port_id	String	Specifies the ID of the NIC port.
mac_addr	String	Specifies the MAC address of the NIC.

Table 8-72 fixed_ips data structure

Parameter	Type	Description
subnet_id	String	Specifies the ID of the subnet (subnet_id) corresponding to the private IP address of the NIC.
ip_address	String	Specifies the NIC private IP address.

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

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.8.2 Querying Information About a Specified BMS NIC (Deprecated)

Function

This API is used to query information about a specified BMS NIC based on the NIC ID.

This API has been deprecated. Use the API in [Querying NICs Bound to a BMS](#).

URI

GET /v2.1/{project_id}/servers/{server_id}/os-interface/{id}

[Table 8-73](#) lists the parameters.

Table 8-73 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
id	Yes	Specifies the NIC ID. You can obtain the NIC ID from the NICs tab page on the BMS details page or by calling the Querying Information About BMS NICs (Deprecated) API. (The NIC ID is the value of port_id).

Request Parameters

None

Example Request

Querying information about a NIC (ID: ce531f90-199f-48c0-816c-13e38010b442) bound to a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)

GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-interface/ce531f90-199f-48c0-816c-13e38010b442

Response Parameters

Parameter	Type	Description
interfaceAttachment	Object	Specifies information about the specified BMS NIC. For details, see Table 8-74 .

Table 8-74 interfaceAttachment data structure

Parameter	Type	Description
port_state	String	Specifies the status of the NIC port. The value can be ACTIVE , BUILD , or DOWN .
fixed_ips	Array of objects	Specifies the IP addresses of NICs. For details, see Table 8-75 .
net_id	String	Specifies the ID of the subnet (network_id) to which the NIC ports belong.
port_id	String	Specifies the ID of the NIC port.
mac_addr	String	Specifies the MAC address of the NIC.

Table 8-75 fixed_ips data structure

Parameter	Type	Description
subnet_id	String	Specifies the ID of the subnet (subnet_id) corresponding to the private IP address of the NIC.
ip_address	String	Specifies the NIC IP address.

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

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.9 BMS Disk Management (OpenStack Nova APIs)

8.9.1 Querying Information About EVS Disks Attached to a BMS (Deprecated)

Function

This API is used to query information about EVS disks attached to a BMS.

This API has been deprecated. Use the API in [Querying EVS Disks Attached to a BMS](#).

URI

GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments

[Table 8-76](#) lists the parameters.

Table 8-76 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying EVS disks attached to a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)

```
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-volume_attachments
```

Response Parameters

Parameter	Type	Description
volumeAttachments	Object	Specifies information about the disks attached to the BMS. For details, see Table 8-77 .

Table 8-77 volumeAttachments data structure

Parameter	Type	Description
device	String	Specifies the mount directory, for example, /dev/vdb .
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ID of the BMS to which the disks are attached.
volumeId	String	Specifies the IDs of the EVS disks attached to the BMS.

Example Response

```
{
  "volumeAttachments": {
    "device": "/dev/vdb",
    "serverId": "820abbd0-2d8e-4bc5-ae46-69cacfd4fbaa",
    "id": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "volumeId": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.9.2 Querying Information About an EVS Disk Attached to a BMS (Deprecated)

Function

This API is used to query information about a single disk attached to a BMS based on the disk ID.

This API has been deprecated. Use the API in [Querying EVS Disks Attached to a BMS](#).

URI

GET /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}

[Table 8-78](#) lists the parameters.

Table 8-78 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
volume_id	Yes	Specifies the EVS disk ID. You can query attached EVS disks attached to a BMS using the Querying Information About EVS Disks Attached to a BMS (Deprecated) API.

Request Parameters

None

Example Request

Querying information about an EVS disk (ID: b53f23bd-ee8f-49ec-9420-d1acfeaf91d6) attached to a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
GET https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-volume_attachments/b53f23bd-ee8f-49ec-9420-d1acfeaf91d6

Response Parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies information about the disk attached to the BMS. For details, see Table 8-79 .

Table 8-79 volumeAttachment data structure

Parameter	Type	Description
device	String	Specifies the mount directory, for example, /dev/vdb .
id	String	Specifies the ID of the attached resource.
serverId	String	Specifies the ID of the BMS to which the disks are attached.
volumeId	String	Specifies the ID of the disk attached to the BMS.

Example Response

```
{
  "volumeAttachment": {
    "device": "/dev/vdb",
    "serverId": "820abbd0-2d8e-4bc5-ae46-69cacfd4fbaa",
    "id": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "volumeId": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.9.3 Attaching an EVS Disk to a BMS (Deprecated)

Function

This API is used to attach an EVS disk to a BMS.

This API has been deprecated. Use the API in [Attaching an EVS Disk to a BMS](#).

Constraints

- A bootable disk cannot be attached to a BMS.
- A disk cannot be attached to a BMS when the BMS is in the **PAUSED** state, which is specified using the **OS-EXT-STS:vm_state** parameter.
- Only a shared disk or a disk in the **available** state can be attached to a BMS.
- Only EVS disks whose device type is **SCSI** can be attached to a BMS.

URI

POST /v2.1/{project_id}/servers/{server_id}/os-volume_attachments

[Table 8-80](#) lists the parameters.

Table 8-80 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
volumeAttachments	Yes	Object	Specifies the disks to be attached. For details, see Table 8-81 .

Table 8-81 volumeAttachment data structure

Parameter	Mandatory	Type	Description
volumeld	Yes	String	Specifies the ID of the disk to be attached to a BMS.
device	No	String	Specifies the mount point, such as /dev/sda and /dev/sdb. The new disk mount point cannot be the same as an existing one. The mount point must be specified based on the sequence of existing device names. Otherwise, the system automatically generates a mount point.

Example Request

Attaching an EVS disk (ID: b53f23bd-ee8f-49ec-9420-d1acfeaf91d6) to a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd) with the mount point of **/dev/sdb**

POST https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-volume_attachments

```
{
  "volumeAttachment": {
    "volumeld": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "device": "/dev/sdb"
  }
}
```

Response Parameters

Parameter	Type	Description
volumeAttachment	Object	Specifies the disks to be attached. For details, see Table 8-82 .

Table 8-82 volumeAttachment data structure

Parameter	Type	Description
device	String	Specifies the device name, for example, /dev/vdb .
serverId	String	Specifies the UUID of the BMS to be attached to.
id	String	Specifies the disk UUID.

Parameter	Type	Description
volumeld	String	Specifies the attaching ID, which is the same as the UUID.

Example Response

```
{
  "volumeAttachment": {
    "id": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "volumeld": "b53f23bd-ee8f-49ec-9420-d1acfeaf91d6",
    "serverId": "820abbd0-2d8e-4bc5-ae46-69cacfd4fbaa",
    "device": "/dev/vdb"
  }
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.9.4 Detaching an EVS Disk from a BMS (Deprecated)

Function

This API is used to detach an EVS disk from a BMS.

This API has been deprecated. Use the API in [Detaching an EVS Disk from a BMS](#).

Constraints

If a BMS is stopped, disks can be detached from it without any limitation on the OS. If a BMS is running, the constraints are as follows:

- Before detaching an EVS disk from a Linux BMS, log in to the BMS, run **umount** to unmount the disk from the file system, and ensure that no program is reading data from or writing data into the disk. Otherwise, the disk will fail to be detached.
- Before detaching an EVS disk from a Windows BMS, ensure that no program is reading data from or writing data into the disk. Otherwise, data will be lost.

URI

DELETE /v2.1/{project_id}/servers/{server_id}/os-volume_attachments/{volume_id}

Table 8-83 lists the parameters.

Table 8-83 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
volume_id	Yes	Specifies the EVS disk ID. You can query attached EVS disks attached to a BMS using the Querying Information About EVS Disks Attached to a BMS (Deprecated) API.

Request Parameters

None

Example Request

Detaching an EVS disk (ID: b53f23bd-ee8f-49ec-9420-d1acfeaf91d6) attached from a BMS (ID: 95bf2490-5428-432c-ad9b-5e3406f869dd)
DELETE https://{ECS Endpoint}/v2.1/c685484a8cc2416b97260938705deb65/servers/95bf2490-5428-432c-ad9b-5e3406f869dd/os-volume_attachments/b53f23bd-ee8f-49ec-9420-d1acfeaf91d6

Response

N/A

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.10 BMS 1D Tag Management (OpenStack Nova APIs)

8.10.1 Querying BMS Tags (Deprecated)

Function

This API is used to query all tags of a BMS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the micro version on the client.

URI

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

[Table 8-84](#) lists the parameters.

Table 8-84 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Querying tags of a BMS (ID: 53206ed0-56de-4d6b-b7ee-ffc62ca26f43)
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/53206ed0-56de-4d6b-b7ee-ffc62ca26f43/tags

Response Parameters

Parameter	Type	Description
tags	Array of strings	Specifies user-defined tags of a BMS.

Example Response

```
{
  "tags": [
    "baz",
    "foo",
    "qux"
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.10.2 Adding Tags for a BMS (Deprecated)

Function

This API is used to add tags for a BMS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the micro version on the client.

Constraints

A BMS can have a maximum of 50 tags.

NOTE

- It is recommended that you add the **__type_baremetal** tag to BMSs to distinguish BMSs from ECSs. Otherwise, BMSs will be available only on the ECS console.
- A new tag will overwrite the existing one. If you want to retain the original tag, add it to the list of new tags. You are advised to add **__type_baremetal** to the added tags list each time you add a tag.

URI

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

[Table 8-85](#) lists the parameters.

Table 8-85 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of strings	- Specifies the tags to be added. Each tag can contain a maximum of 80 characters. - The tag cannot start with a period (.). - A BMS can have a maximum of 50 tags. - An empty tag cannot be created.

Example Request

Adding tags **baz**, **foo**, and **qux** for a BMS (ID: 53206ed0-56de-4d6b-b7ee-ffc62ca26f43)

PUT https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/53206ed0-56de-4d6b-b7ee-ffc62ca26f43/tags

```
{
  "tags": [
    "baz",
    "foo",
    "qux"
  ]
}
```

Response Parameters

Parameter	Type	Description
tags	Array of strings	Specifies user-defined tags of a BMS.

Example Response

```
{
  "tags": [
    "baz",
    "foo",
    "qux"
  ]
}
```

Returned Values

Normal values

Returned Values	Description
200	The request has been successfully processed.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.10.3 Deleting Tags from a BMS (Deprecated)

Function

This API is used to delete all tags of a BMS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the micro version on the client.

Constraints

NOTE

- Tag **__type_baremetal** is used to identify a BMS. You are not advised to delete this tag. Otherwise, the BMS will be displayed only on the ECS console.
- After deleting the **__type_baremetal** tag, you can add it again by following the instructions in [Adding a Tag for a BMS \(Deprecated\)](#). After the tag is added, the BMS will be displayed on the BMS console.

URI

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

[Table 8-86](#) lists the parameters.

Table 8-86 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Request Parameters

None

Example Request

Deleting tags from a BMS (ID: 53206ed0-56de-4d6b-b7ee-ffc62ca26f43)

```
DELETE https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/53206ed0-56de-4d6b-b7ee-ffc62ca26f43/tags
```

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.10.4 Adding a Tag for a BMS (Deprecated)

Function

This API is used to add a tag for a BMS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the micro version on the client.

Constraints

- A BMS can have a maximum of 50 tags.
- The tag contains a maximum of 80 characters.
- The tag cannot start with a period (.).
- An empty tag cannot be created.

NOTE

It is recommended that you add the **__type_baremetal** tag to BMSs to distinguish BMSs from ECSs.

URI

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

[Table 8-87](#) lists the parameters.

Table 8-87 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
tag	Yes	Specifies the tag information. Constraints: • The tag contains a maximum of 80 characters. • The tag cannot start with a period (.). • An empty tag cannot be created. • URL encoding is required for special characters.

Request Parameters

None

Example Request

Adding a tag for a BMS (ID: 53206ed0-56de-4d6b-b7ee-ffc62ca26f43)
PUT https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/53206ed0-56de-4d6b-b7ee-ffc62ca26f43/tags/{tag}

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.10.5 Checking for Tags of a BMS (Deprecated)

Function

This API is used to check whether a BMS has a specified tag.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the micro version on the client.

URI

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

[Table 8-88](#) lists the parameters.

Table 8-88 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .
tag	Yes	Specifies the key of the tag to be queried. Constraints: • URL encoding is required for special characters. • If no tag key is specified, all tags of the BMS are displayed.

Request Parameters

None

Example Request

Checking whether a BMS (ID: 2d85af7c-cbfe-40c5-a378-4d03b42fb0e2) has a specified tag

```
GET https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/2d85af7c-cbfe-40c5-a378-4d03b42fb0e2/tags/{tag}
```

Example Response

If the specified tag exists, no response is returned.

If the specified tag does not exist, the response is as follows:

```
{
  "itemNotFound": {
    "message": "Server 2d85af7c-cbfe-40c5-a378-4d03b42fb0e2 has no tag 'abc'",
    "code": 404
  }
}
```

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

8.10.6 Deleting a Tag from a BMS (Deprecated)

Function

This API is used to delete a tag of a BMS.

You are required to use the HTTP header **X-OpenStack-Nova-API-Version: 2.26** to specify the micro version on the client.

Constraints

- The tag contains a maximum of 80 characters.
- If a tag contains non-URL-safe characters, perform URL encoding.

NOTE

- Tag **__type_baremetal** is used to identify a BMS. You are not advised to delete this tag. Otherwise, the BMS will be displayed only on the ECS console.
- After deleting the **__type_baremetal** tag, you can add it again by following the instructions in [Adding a Tag for a BMS \(Deprecated\)](#). After the tag is added, the BMS will be displayed on the BMS console.

URI

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

[Table 8-89](#) lists the parameters.

Table 8-89 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see Obtaining a Project ID .
server_id	Yes	Specifies the BMS ID. You can obtain the BMS ID from the BMS console or by calling the API Querying BMSs (Deprecated) .

Parameter	Mandatory	Description
tag	Yes	Specifies the tag information. Constraints: • A tag can contain a maximum of 80 characters. If a tag contains non-URL-safe characters, perform URL encoding. • If no key is specified, all tags of the BMS are deleted.

Request Parameters

None

Example Request

Deleting a tag from a BMS (ID: 53206ed0-56de-4d6b-b7ee-ffc62ca26f43)
DELETE https://{ECS Endpoint}/v2.1/bbf1946d374b44a0a2a95533562ba954/servers/53206ed0-56de-4d6b-b7ee-ffc62ca26f43/tags/{tag}

Response

N/A

Returned Values

Normal values

Returned Values	Description
204	The server has processed the request but did not return any content.

For details about other returned values, see [Status Codes](#).

Error Codes

See [Error Codes](#).

A Appendix

A.1 Obtaining a Project ID

Scenarios

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

- [Obtain the Project ID by Calling an API](#)
- [Obtain the Project ID from the Console](#)

Obtain the Project ID by Calling an API

You can obtain a project ID by calling the API used to [query projects based on specified criteria](#).

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

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

```
{
  "projects": [
    {
      "domain_id": "65ewtrgaggshhk1223245sghjlse684b",
      "is_domain": false,
      "parent_id": "65ewtrgaggshhk1223245sghjlse684b",
      "name": "project_name",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects/a4adasfjljaaakla12334jklga9sasfg"
      },
      "id": "a4adasfjljaaakla12334jklga9sasfg",
      "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.

A.2 Obtaining an Account ID

An account ID is required for some URLs when an API is called. To obtain an account ID, 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 **My Credentials** page, view the **Account ID**.