

Dedicated Computing Cluster

API Reference

Issue 01
Date 2022-08-10



Copyright © Huawei Technologies Co., Ltd. 2024. 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	4
3.1 Making an API Request	4
3.2 Authentication	8
3.3 Response	9
4 DCC APIs	12
4.1 Querying DCC Resources	12
5 Common Parameters	16
5.1 Status Codes	16
5.2 Obtaining a Project ID	17

1 Before You Start

Dedicated Computing Cluster (DCC) provides dedicated, physically isolated computing resource pools on the cloud service platform, allowing you to use physical computing devices and resources exclusively. You can apply for a host machine in your resource pool to host your own Elastic Cloud Servers (ECSs).

This document describes how to use application programming interfaces (APIs) to perform operations on , such as creating, querying, deleting, and modifying . For details about all supported operations, see [API Overview](#).

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

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

Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions.

Before you use an API to call resources, specify its region and endpoint. For more details, see [Regions and Endpoints](#).

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**

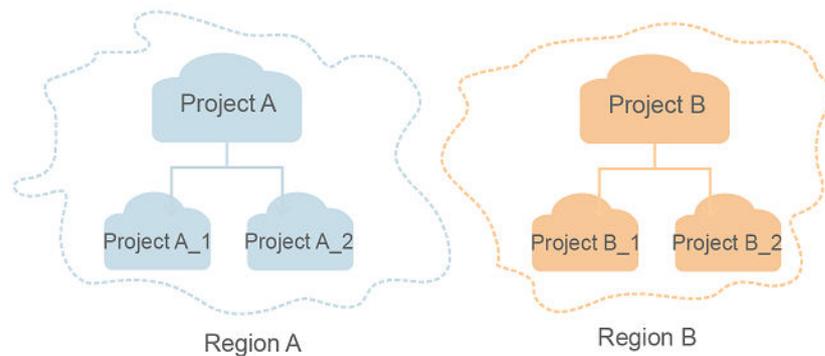
Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. 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 to group and physically isolate resources (including computing, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolation model



- Enterprise project**

Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated. An enterprise project can contain resources of multiple projects, and resources can be added to or removed from enterprise projects.

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

2 API Overview

Table 2-1 APIs

API	Description
Querying Resources	Query the resources that you have obtained.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a RESTful 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.

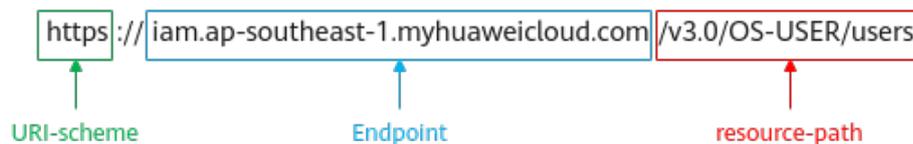
Table 3-1 URI parameter description

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 CN-Hong Kong is iam.ap-southeast-1.myhuaweicloud.com .
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is /v3/auth/tokens .
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 **CN-Hong Kong** region, obtain the endpoint of IAM (**iam.ap-southeast-1.myhuaweicloud.com**) 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.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users`

Figure 3-1 Example URI



NOTE

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

Request Methods

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

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Same as GET except that the server must return only the response header.
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.ap-southeast-1.myhuaweicloud.com/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 listed in [Table 3-3](#).

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	<p>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).</p> <p>After the request is processed, the value of X-Subject-Token in the response header is the token value.</p>	<p>No</p> <p>This field is mandatory for token authentication.</p>	<p>The following is part of an example token:</p> <p>MIIPAgYJKoZlhvc NAQcCo...ggg1B BIINPXsidG9rZ</p>

 **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.ap-southeast-1.myhuaweicloud.com/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.ap-southeast-1.myhuaweicloud.com/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"  
  }  
}
```

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

3.2 Authentication

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

- 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: Requests are authenticated using tokens.

AK/SK Authentication

NOTE

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

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

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

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

NOTE

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

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.

IMS 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.ap-southeast-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

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 3-2 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 3-2 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 is 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 DCC APIs

4.1 Querying DCC Resources

Function

This API is used to query the DCC resources that you have obtained.

URI

GET /v1/{project_id}/dcc/resource_clusters

[Table 4-1](#) describes the parameters.

Table 4-1 Parameters

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

Request Parameters

You can add the **service_type** parameter to the URI to filter the query result.

For example: /v1/{project_id}/dcc/resource_clusters?
service_type={service_type}

Parameter	Type	Mandatory	Description
service_type	String	No	Specifies the cluster service type. The value can be ecs or bms .

Example Request

Querying the obtained resources whose **service_type** is **ecs**

GET https://{Endpoint}/v1/056f7ec7e100d3d32f81c014630b5648/dcc/resource_clusters?service_type=ecs

Response Parameters

Table 4-2 Response parameters

Parameter	Type	Description
dedicated_cluster	Array of objects	Specifies the provisioned . For details, see Table 4-3 .

Table 4-3 dedicated_cluster field description

Parameter	Type	Description
project_id	String	Specifies the project ID.
availability_zone	String	Specifies the AZ.
host_total	Integer	Specifies the number of hosts.
host_type	String	Specifies the host specifications code.
service_type	String	Specifies the cluster service type. The value can be ecs or bms .
host_properties	Object	Specifies the host specifications.
vcpus_used	Integer	Specifies the number of used vCPUs.
vcpus_total	Integer	Specifies the total number of vCPUs.
memory_mb_used	Integer	Specifies the used memory size.
memory_mb_total	Integer	Specifies the total memory size.
flavors	Array of strings	Specifies the supported flavors.
instance_total	Integer	Specifies the total number of running compute instances.

Table 4-4 host_properties field description

Parameter	Type	Description
cpu_cores	Integer	Specifies the number of cores.
memory_mb	Integer	Specifies the memory size.

Parameter	Type	Description
cpu_speed	String	Specifies the CPU frequency.

Example response

```
{
  "dedicated_cluster": [
    {
      "project_id": "056f7ec7e100d3d32f81c014630b5648",
      "availability_zone": "kvmxen.dc1",
      "host_total": 1,
      "host_type": "dec_IOOptimizedS2",
      "service_type": "ecs",
      "host_properties": {
        "cpu_cores": 216,
        "memory_mb": 264192,
        "cpu_speed": "2.5"
      },
      "vcpus_used": 1,
      "vcpus_total": 96,
      "memory_mb_used": 2048,
      "memory_mb_total": 264192,
      "flavors": [
        "s2.2xlarge.2",
        "s2.2xlarge.4",
        "s2.4xlarge.2",
        "s2.4xlarge.4",
        "s2.8xlarge.2",
        "s2.8xlarge.4",
        "s2.large.2",
        "s2.large.4",
        "s2.medium.2",
        "s2.medium.4",
        "s2.xlarge.2",
        "s2.xlarge.4"
      ],
      "instance_total": 1
    },
    {
      "project_id": "056f7ec7e100d3d32f81c014630b5648",
      "availability_zone": "kvmxen.dc1",
      "host_total": 3,
      "host_type": "dec_IOOptimizedS2",
      "hypervisor_type": "ecs",
      "host_properties": {
        "cpu_cores": 108,
        "memory_mb": 264192,
        "cpu_speed": "2.5"
      },
      "vcpus_used": 24,
      "vcpus_total": 134,
      "memory_mb_used": 53248,
      "memory_mb_total": 792576,
      "flavors": [
        "s2.2xlarge.2",
        "s2.2xlarge.4",
        "s2.4xlarge.2",
        "s2.4xlarge.4",
        "s2.8xlarge.2",
        "s2.8xlarge.4",
        "s2.large.2",
        "s2.large.4",
        "s2.medium.2",
        "s2.medium.4",
        "s2.xlarge.2",
        "s2.xlarge.4"
      ]
    }
  ]
}
```

```
        "s2.xlarge.4"  
      ],  
      "instance_total": 23  
    }  
  ]  
}
```

Status Code

See [Status Codes](#).

5 Common Parameters

5.1 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.

5.2 Obtaining a Project ID

Scenarios

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

- [Obtain the Project ID by Calling an API](#)
- [Obtain the Project ID from the Console](#)

Obtain the Project ID by Calling an API

You can obtain a project ID by calling the API used to [query projects based on specified criteria](#).

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

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

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "project_name",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
      }
    }
  ]
}
```

```
    "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"  
  },  
  "id": "a4a5d4098fb4474fa22cd05f897d6b99",  
  "enabled": true  
}  
],  
"links": {  
  "next": null,  
  "previous": null,  
  "self": "https://www.example.com/v3/projects"  
}  
}
```

Obtain a Project ID from the Console

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

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

Figure 5-1 Viewing the project ID

