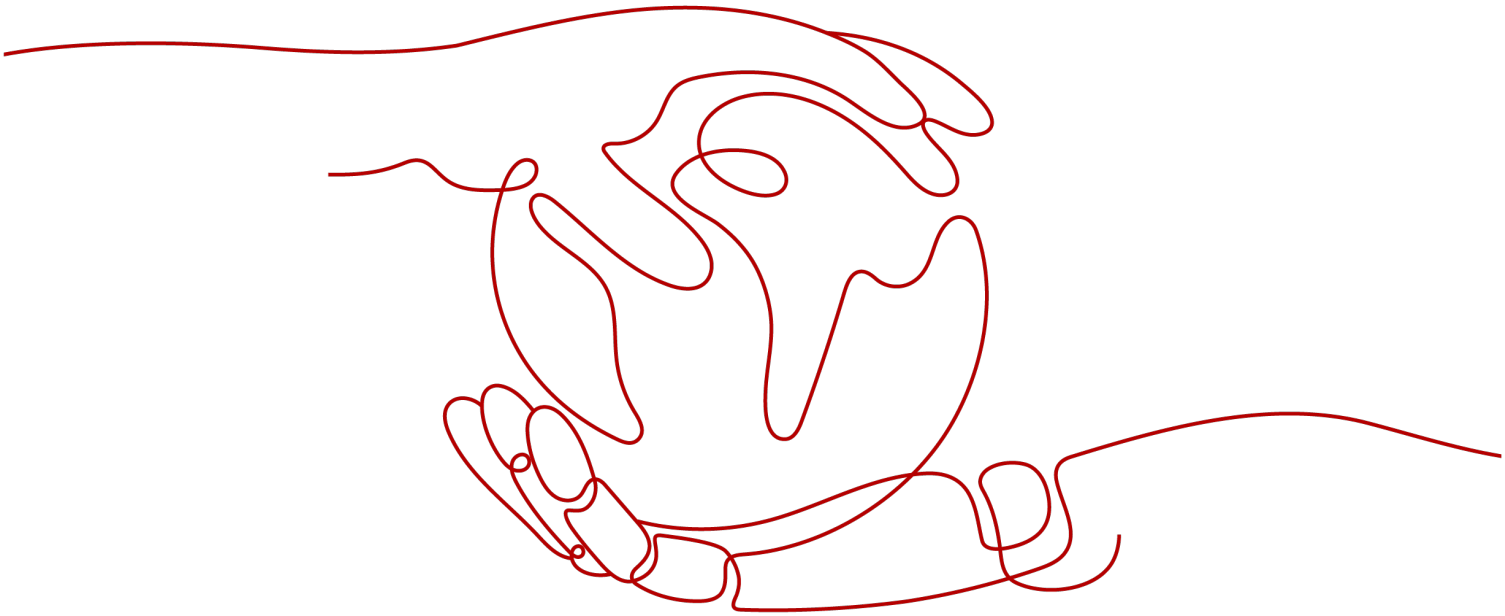
Distributed Message Service for Kafka

API Reference

Issue 01

Date 2022-08-12



HUAWEI

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Contents

1 Before You Start

1.1 Overview

Welcome to *Distributed Message Service for Kafka API Reference*. DMS for Kafka is a message queuing service that uses the open-source Apache Kafka. It provides Kafka instances with isolated computing, storage, and bandwidth resources. DMS for Kafka allows you to apply resources, configure topics, partitions, and replicas based on service requirements. The service can be used out of the box and frees you from deployment and O&M so that you can focus on the agile development of your applications.

This document describes functions, syntax, parameters, and examples of the application programming interfaces (APIs) of DMS for Kafka.

NOTICE

DMS for Kafka is continuously upgraded with new functions, and the existing APIs are inevitably adjusted. For example, new response parameters are added.

To reduce the impact of API changes, DMS for Kafka is backward compatible with existing APIs. When using DMS, you should accept and ignore unused parameters and parameter values in JSON responses.

1.2 API Calling

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

1.3 Endpoints

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

1.4 Constraints

- The number of instances that you can create is determined by your quota. For details, see **Service Quota**.
- For more constraints, see API description.

1.5 Concepts

- Account

  An 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 and should not be used directly to perform routine management. For security purposes, create IAM users and grant them permissions for routine management.

- IAM user

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

  The account name, username, and password will be required for API authentication.

- Region: A region is a geographic area in which cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other over an intranet, while AZs in different regions are isolated from each other. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.
- An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, network, storage, and other resources 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

  Projects group and isolate resources (including compute, storage, and network resources) across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. For more refined access control, create subprojects under a project and purchase resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.

- Enterprise project

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

2 API Overview

Table 2-1 APIs for managing Kafka premium instances

API	Description
Creating an Instance	Creates instances.
Querying an Instance	Queries detailed information about an instance.
Modifying an Instance	Modifies the name and description of an instance.
Deleting an Instance	Deletes a specified instance and release all the resources occupied by it.
Restarting or Deleting Instances in Batches	Restarts or deletes instances in batches.
Querying All Instances	Queries the instances of a tenant by set conditions.
Querying AZ Information	Queries AZ information.
Querying Product Specifications	Queries product specifications.
Querying Maintenance Time Windows	Queries maintenance time windows.
Creating a Topic in a Kafka Instance	Creates a topic in an instance.
Querying a Topic in a Kafka Instance	Queries details about a topic in an instance.
Deleting Topics in a Kafka Instance in Batches	Deletes topics in an instance in batches.

3 Calling APIs

3.1 Making an API Request

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

Request URI

A request URI is in the following format:

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

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

- **URI-scheme**:

  Protocol used to transmit requests. All APIs use HTTPS.
- **Endpoint**:

  Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from **Regions and Endpoints**.
- **resource-path**:

  Access path of an API for performing a specified operation. Obtain the value from the URI of the 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 a query parameter that is in the format of "*Parameter name=Parameter value*". For example, **? limit=10** indicates that a maximum of 10 data records will be displayed.

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:

- **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**: requests a server resource header.
- **PATCH**: requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

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

```
POST https://iam.my-kualalumpur-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

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

Common request headers are as follows:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json**. Other values of this field will be provided for specific APIs if any.
- **X-Auth-Token**: specifies a user token only for token-based API authentication. The user token is a response to the API used to **obtain a user token**. This API is the only one that does not require authentication.

  NOTE

  In addition to supporting token-based authentication, DMS APIs also support authentication using access key ID/secret access key (AK/SK). During AK/SK-based authentication, an SDK is used to sign the request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

  For more information, see **AK/SK-based Authentication**.

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

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

Request Body

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

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

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

NOTE

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

```
POST https://{{endpoint}}/v3/auth/tokens
Content-Type: application/json

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

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

3.2 Authentication

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

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

Token-based Authentication

NOTE

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

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

A cloud service can be deployed as either a project-level service or global service.

- For a project-level service, you need to obtain a project-level token. When you call the API, set **auth.scope** in the request body to **project**.
- For a global service, you need to obtain a global token. When you call the API, set **auth.scope** in the request body to **domain**.

DMS is a project-level service. When calling a DMS API, a project-level token is required. Therefore, when you call the API used to **obtain a user token**, set **auth.scope** in the request body to **project**.

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

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

```
POST https://{{endpoint}}/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

NOTE

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

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

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

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

NOTICE

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 the 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 Code**.

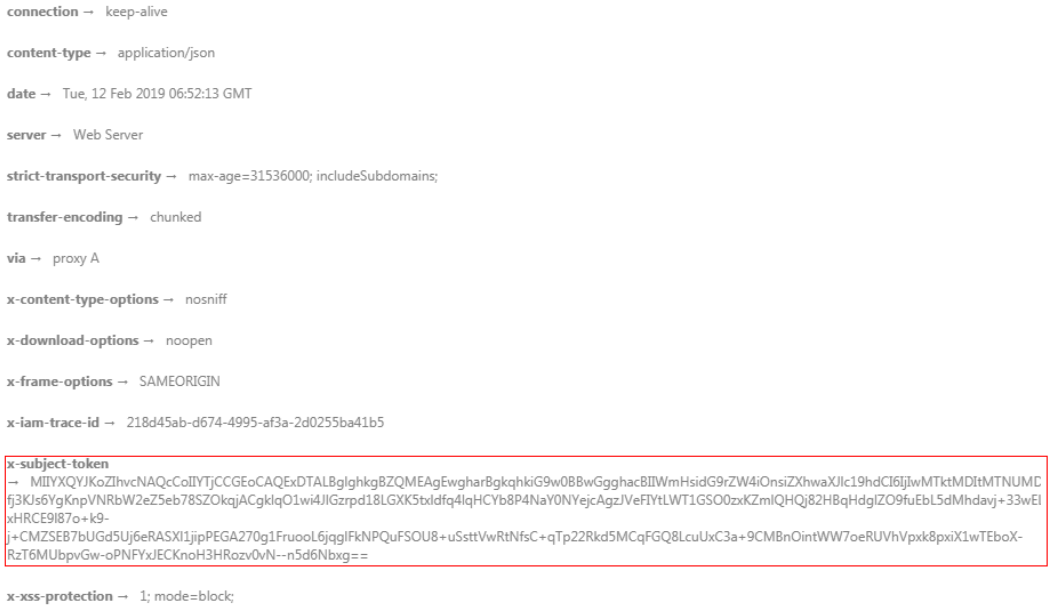
For example, if status code **201** is returned for calling the API used to **obtain a user token**, the request is successful.

Response Header

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

Figure 3-1 shows the response header for the API of **obtaining a user token**, in which **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

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



(Optional) Response Body

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

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

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

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

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

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

4 Getting Started

Scenarios

This section describes how to call an API to create a Kafka instance and customize the computing capabilities and storage space of the instance based on your service requirements.

For details on how to call APIs, see **Calling APIs**.

Prerequisites

- IAM endpoint obtained from **Regions and Endpoints**.
- Kafka endpoint obtained from **Regions and Endpoints**.

Creating a Kafka Instance

The following is an example request for creating a Kafka instance:

```
{
   "name": "kafka-demo",
   "engine": "kafka",
   "engine_version": "2.3.0",
   "storage_space": 600,
   "kafka_manager_user": "******",
   "kafka_manager_password": "******",
   "vpc_id": "1a28dcc5-c90d-421c-82bb-783f30f5b40a",
   "security_group_id": "0cc8fdb7-872a-49da-a062-88ccc39463b5",
   "subnet_id": "ebba7994-260d-42ab-bce1-39a08b365dc8",
   "available_zones": ["ae04cf9d61544df3806a3feeb401b204"],
   "product_id": "00300-30308-0--0",
   "specification": "100MB",
   "storage_spec_code": "dms.physical.storage.ultra",
   "partition_num": "300"
}
```

- **name**: name of the instance.
- **engine**: message engine. The value is **kafka**.
- **engine_version**: version of the message engine.
- **storage_space**: message storage space in GB. For details about the value range, see **Table 5-2**.
- **kafka_manager_user**: username for logging in to Kafka Manager
- **kafka_manager_password**: password for logging in to Kafka Manager

- **vpc_id**: ID of the Virtual Private Cloud (VPC) where the instance resides. For details, see *Virtual Private Cloud API Reference*.
- **security_group_id**: ID of the security group. *For details, see Virtual Private Cloud API Reference.*
- **subnet_id**: ID of the VPC subnet. *For details, see Virtual Private Cloud API Reference.*
- **available_zones**: ID of the AZ where the instance resides. The value cannot be empty or null. For details, see **Querying AZ Information**.
- **product_id**: ID of the product. For details, see **Querying Product Specifications**.
- **specification**: bandwidth of the instance. For details about the value range, see **Table 5-2**.
- **storage_spec_code**: storage I/O specification. For details about the value range, see **Table 5-2**.
- **partition_num**: maximum partitions in the instance. For details about the value range, see **Table 5-2**.

5 APIs for Managing Instances

5.1 Creating an Instance

Function

This API is used to create a Kafka instance.

URI

POST /v1.0/{project_id}/instances

Table 5-1 describes the parameter.

Table 5-1 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.

Request

Request parameters

Table 5-2 describes the parameters.

Table 5-2 Request parameters

Parameter	Type	Mandatory	Description
name	String	Yes	Instance name. An instance name starts with a letter, consists of 4 to 64 characters, and can contain letters, digits, and hyphens (-).
description	String	No	Description of an instance. It is a character string containing not more than 1024 characters. **NOTE** The backslash (\) and quotation mark (") are special characters for JSON packets. When using these characters in a parameter value, add the escape character (\) before these characters, for example, **** and **\"**.
engine	String	Yes	Message engine. Set the value to **kafka**.
engine_version	String	Yes	Version of the message engine. Value: **1.1.0** or **2.3.0**.
specification	String	Yes	Baseline bandwidth of a Kafka instance, that is, the maximum amount of data transferred per unit time. Unit: MB Options: ● 100MB ● 300MB ● 600MB ● 1200MB
storage_space	Integer	Yes	Message storage space. Unit: GB. Value range: ● Kafka instance with **specification** being **100MB**: 600–90,000 GB ● Kafka instance with **specification** being **300MB**: 1200–90,000 GB ● Kafka instance with **specification** being **600MB**: 2400–90,000 GB ● Kafka instance with **specification** being **1200MB**: 4800–90,000 GB

Parameter	Type	Mandatory	Description
partition_num	Integer	Yes	Maximum number of partitions in a Kafka instance. Options: ● When **specification** is **100MB**: **300** ● When **specification** is **300MB**: **900** ● When **specification** is **600MB**: **1800** ● When **specification** is **1200 MB**: **1800**
access_user	String	No	This parameter is mandatory when **ssl_enable** is set to **true**. This parameter is invalid when **ssl_enable** is set to **false**. Username. A username consists of 4 to 64 characters and can contain letters, digits, and hyphens (-).
password	String	No	This parameter is mandatory when **ssl_enable** is set to **true**. This parameter is invalid when **ssl_enable** is set to **false**. Instance password. The password must meet the following complexity requirements: ● Must be a string consisting of 8 to 32 characters. ● Contains at least three of the following characters: – Lowercase letters – Uppercase letters – Digits – Special characters `~!@#$%^&*()-_=+|[{}];:',<.>/?
vpc_id	String	Yes	VPC ID. Obtain the value by using either of the following methods: ● Method 1: Log in to VPC console and view the VPC ID in the VPC details. ● Method 2: Call the API for querying VPCs. For details, see the "Querying VPCs" section in the *Virtual Private Cloud API Reference*.

Parameter	Type	Mandatory	Description
security_group_id	String	Yes	Security group which the instance belongs to. Obtain the value by using either of the following methods: • Method 1: Log in to VPC console. Choose **Access Control** > **Security Groups** in the navigation pane on the left. On the displayed page, click the target security group. You can view the security group ID on the displayed page. • Method 2: Call the API for querying security groups. For details, see the "Querying Security Groups" section in the *Virtual Private Cloud API Reference*.
subnet_id	String	Yes	Subnet ID. Obtain the value by using either of the following methods: • Method 1: Log in to VPC console and click the target subnet on the **Subnets** tab page. You can view the network ID on the displayed page. • Method 2: Call the API for querying subnets. For details, see the "Querying Subnets" section in the *Virtual Private Cloud API Reference*.
available_zones	Array	Yes	ID of the AZ where brokers reside and which has available resources. The parameter value cannot be an empty array or an empty array. For details on how to obtain the value, see **Querying AZ Information**. Check whether the AZ has available resources. When creating a Kafka instance, you can select either 1 AZ or at least 3 AZ. When specifying AZs for brokers, use commas (,) to separate multiple AZs.
product_id	String	Yes	Product ID. For details on how to obtain the ID, see **Querying Product Specifications**.
kafka_manager_user	String	Yes	Username for logging in to the Kafka Manager. The username consists of 4 to 64 characters and can contain letters, digits, hyphens (-), and underscores (_).

Parameter	Type	Mandatory	Description
kafka_manager_password	String	Yes	Password for logging in to the Kafka Manager. The password must meet the following complexity requirements: • Must be a string consisting of 8 to 32 characters. • Contains at least three of the following characters: – Lowercase letters – Uppercase letters – Digits – Special characters `~!@#$%^&*()-_=+|[{}];:',<.>/?
maintain_begin	String	No	Time at which a maintenance time window starts. Format: HH:mm:ss • The start time and end time of the maintenance time window must indicate the time segment of a supported maintenance time window. For details about how to query the time segments of supported maintenance time windows, see **Querying Maintenance Time Windows**. • The start time must be set to 22:00:00, 02:00:00, 06:00:00, 10:00:00, 14:00:00, or 18:00:00. • Parameters **maintain_begin** and **maintain_end** must be set in pairs. If parameter **maintain_begin** is left blank, parameter **maintain_end** is also left blank. In this case, the system automatically sets the start time to 02:00:00.

Parameter	Type	Mandatory	Description
maintain_end	String	No	Time at which a maintenance time window ends. Format: HH:mm:ss • The start time and end time of the maintenance time window must indicate the time segment of a supported maintenance time window. For details about how to query the time segments of supported maintenance time windows, see **Querying Maintenance Time Windows**. • The end time is four hours later than the start time. For example, if the start time is 22:00:00, the end time is 02:00:00. • Parameters **maintain_begin** and **maintain_end** must be set in pairs. If parameter **maintain_end** is left blank, parameter **maintain_start** is also blank. In this case, the system automatically sets the end time to 06:00:00.
ssl_enable	Boolean	No	Whether to enable SSL-encrypted access. • **true**: enable • **false**: disable
retention_policy	String	No	Action to be taken when the memory usage reaches the disk capacity threshold. Options: • **time_base**: Automatically delete the earliest messages. • **produce_reject**: Stop producing new messages.
enable_auto_topic	Boolean	No	Whether to enable automatic topic creation. • **true**: enable • **false**: disable If automatic topic creation is enabled, a topic will be automatically created with 3 partitions and 3 replicas when a message is produced to or consumed from a topic that does not exist.

Parameter	Type	Mandatory	Description
storage_spec_code	String	Yes	Storage I/O specification. Options: • **dms.physical.storage.high** or **dms.physical.storage.ultra** when the parameter **specification** is **100MB** • **dms.physical.storage.high** or **dms.physical.storage.ultra** when the parameter **specification** is **300MB** • **dms.physical.storage.ultra** when the parameter **specification** is **600MB** • **dms.physical.storage.ultra** when the parameter **specification** is **1200MB**
enterprise_project_id	String	No	Enterprise project ID.
tags	Array<Object>	No	List of tags.

Table 5-3 tags

Parameter	Type	Mandatory	Description
key	String	No	Tag key. A tag key can contain a maximum of 36 Unicode characters. The key cannot be left blank or be an empty string. It cannot contain nonprintable ASCII (0–31) characters and the following special characters: =*<>\,|/
value	String	No	Tag value. A tag value can contain a maximum of 43 Unicode characters. The value cannot be left blank or be an empty string. It cannot contain nonprintable ASCII (0–31) characters and the following special characters: =*<>\,|/

```
{
   "name": "kafka-test",
   "description": "",
   "engine": "kafka",
   "engine_version": "XXXX",
```



```
  "storage_space": 600,
  "access_user": "",
  "password": "",
  "kafka_manager_user": "******",
  "kafka_manager_password": "******",
  "vpc_id": "1e93f86e-13af-46c8-97d6-d40fa62b76c2",
  "security_group_id": "0aaa0033-bf7f-4c41-a6c2-18cd04cad2c8",
  "subnet_id": "b5fa806c-35e7-4299-b659-b39398dd4718",
  "available_zones": ["d573142f24894ef3bd3664de068b44b0"],
  "product_id": "00300-30308-0--0",
  "maintain_begin": "22:00:00",
  "maintain_end": "02:00:00",
  "ssl_enable": false,
  "enterprise_project_id": "0",
  "specification": "100MB",
  "partition_num": "300",
  "retention_policy": "produce_reject",
     "enable_auto_topic": true,
  "storage_spec_code": "dms.physical.storage.ultra"
}
```

Response

Response parameters

Table 5-4 describes the parameters.

Table 5-4 Response parameters

Parameter	Type	Description
instance_id	String	Instance ID.

Example response

```
{
  "instance_id": "8959ab1c-7n1a-yyb1-a05t-93dfc361b32d"
}
```

Status Code

Table 5-5 describes the status code of successful operations. For details about other status codes, see **Status Code**.

Table 5-5 Status code

Status Code	Description
200	The instance is created successfully.

5.2 Querying an Instance

Function

This API is used to query the details about an instance.

URI

GET /v1.0/{project_id}/instances/{instance_id}

Table 5-6 describes the parameters.

Table 5-6 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.
instance_id	String	Yes	Instance ID.

Request

Request parameters

None.

Example request

None.

Response

Response parameters

Table 5-7 describes the parameters.

Table 5-7 Response parameters

Parameter	Type	Description
name	String	Instance name.
engine	String	Message engine.
engine_version	String	Version of the message engine.
specification	String	Instance specification.
storage_space	Integer	Message storage space. Unit: GB.
partition_num	String	Total number of partitions in a Kafka instance.

Parameter	Type	Description
used_storage_space	Integer	Used message storage space. Unit: GB
connect_address	String	IP address of an instance.
port	Integer	Port number of an instance.
status	String	Instance status. For details, see **Instance Status**.
instance_id	String	Instance ID.
resource_spec_code	String	Resource specifications identifier. • **dms.instance.kafka.cluster.c3.mini**: Kafka instance, 100 MB/s reference bandwidth • **dms.instance.kafka.cluster.c3.small.2**: Kafka instance, 300 MB/s reference bandwidth • **dms.instance.kafka.cluster.c3.middle.2**: Kafka instance, 600 MB/s reference bandwidth • **dms.instance.kafka.cluster.c3.high.2**: Kafka instance, 1200 MB/s reference bandwidth
type	String	Instance type. Value: **cluster**
charging_mode	Integer	Billing mode.
vpc_id	String	ID of a VPC.
vpc_name	String	Name of a VPC.
created_at	String	Time when the instance is created. The time is in the format of timestamp, that is, the offset milliseconds from 1970-01-01 00:00:00 UTC to the specified time.
product_id	String	Product ID.
security_group_id	String	Security group ID.
security_group_name	String	Security group name.
subnet_id	String	Subnet ID.
subnet_name	String	Subnet name.
subnet_cidr	String	Subnet segment.
available_zones	Array	ID of the AZ to which the instance node belongs. The AZ ID is returned.
user_id	String	User ID.

Parameter	Type	Description
user_name	String	Username.
access_user	String	Username of an instance.
maintain_begin	String	Time at which a maintenance time window starts. Format: HH:mm:ss
maintain_end	String	Time at which a maintenance time window ends. Format: HH:mm:ss
management_connect_address	String	Connection address of Kafka Manager of the Kafka instance.
ssl_enable	Boolean	Whether security authentication is enabled. • **true**: enable • **false**: disable
enterprise_project_id	String	Enterprise project ID.
is_logical_volume	Boolean	Distinguishes old instances from new instances during instance capacity expansion. • **true**: New instance, which allows dynamic disk capacity expansion without restarting the instance. • **false**: Old instance.
extend_times	Integer	Number of disk expansion times. If it exceeds 20, the disk cannot be expanded.
enable_auto_topic	Boolean	Whether automatic topic creation is enabled. • **true**: enabled • **false**: disabled
total_storage_space	Integer	Message storage space. Unit: GB
storage_resource_id	String	Storage resource ID.
storage_spec_code	String	I/O specification.
service_type	String	Service type.
storage_type	String	Storage type.
retention_policy	String	Message retention policy.
kafka_manager_user	String	Username for logging in to Kafka Manager.

Parameter	Type	Description
enable_log_collection	Boolean	Whether log collection is enabled.
cross_vpc_info	String	Cross-VPC access information.
ipv6_enable	Boolean	Whether IPv6 is enabled.
ipv6_connect_addresses	Array of strings	IPv6 connection address.
rest_enable	Boolean	Whether the Kafka REST function is enabled.
rest_connect_address	String	Kafka REST connection address.
message_query_inst_enable	Boolean	Whether message query is enabled.
vpc_client_plain	Boolean	Whether intra-VPC plaintext access is enabled.
support_features	String	List of features supported by the Kafka instance.
trace_enable	Boolean	Whether message tracing is enabled.
pod_connect_address	String	Connection address on the tenant side.
disk_encrypted	Boolean	Whether disk encryption is enabled.
kafka_private_connect_address	String	Private connection address of a Kafka instance.
ces_version	String	Cloud Eye version.
tags	Array<Object>	List of tags.

Table 5-8 tags

Parameter	Type	Description
key	String	Tag key.
value	String	Tag value.

Example response

```
{
    "name": "kafka-l00230526",
    "engine": "kafka",
    "port": 9092,
    "status": "RUNNING",
    "type": "cluster",
```



```
    "specification": "100MB",
    "engine_version": "XXX",
    "connect_address": "192.168.1.116,192.168.1.152,192.168.1.78",
    "connect_dn": "",
    "instance_id": "ef84dd5f-3ece-4336-8c99-987defd62e3a",
    "resource_spec_code": "dms.instance.kafka.cluster.c3.mini",
    "charging_mode": 1,
    "vpc_id": "2477879f-aebf-496f-a08a-67812885ce9b",
    "vpc_name": "vpc-y00502467",
    "created_at": "1568797295209",
    "product_id": "00300-30308-0--0",
    "security_group_id": "008a08e2-10cc-4d9b-90ab-3f3b8f6c3333",
    "security_group_name": "z00417080-cce-node-na7j",
    "subnet_id": "5ca08fb7-7522-4d95-9fa5-ff6b3592a29d",
    "subnet_name": "subnet-cyd-6102",
    "subnet_cidr": "192.168.1.0/24",
    "available_zones": [
        "ae04cf9d61544df3806a3feeb401b204"
    ],
    "user_id": "2b4af4428ec840dfa1f0f1a32e965567",
    "user_name": "laiyh",
    "kafka_manager_user": "root",
    "maintain_begin": "22:00:00",
    "maintain_end": "02:00:00",
    "storage_space": 492,
    "total_storage_space": 600,
    "used_storage_space": 25,
    "partition_num": "300",
    "ssl_enable": false,
    "management_connect_address": "https://192.168.1.116:9999",
    "storage_resource_id": "81982562-ce8b-490a-95fa-2b225c292271",
    "storage_spec_code": "dms.physical.storage.ultra",
    "service_type": "advanced",
    "storage_type": "hec",
    "enterprise_project_id": "0",
    "is_logical_volume": true,
    "extend_times": 0,
    "retention_policy": "produce_reject",
    "ipv6_enable": false,
    "ipv6_connect_addresses": [],
    "connector_enable": false,
    "connector_id": "",
    "rest_enable": false,
    "rest_connect_address": "",
    "message_query_inst_enable": true,
    "vpc_client_plain": false,
    "support_features":
"feature.physerver.kafka.topic.accesspolicy,message_trace_enable,features.pod.token.access,feature.physerver.
kafka.pulbic.dynamic,feature.physerver.kafka.user.manager",
    "trace_enable": false,
    "agent_enable": false,
    "pod_connect_address": "100.113.16.105:9100,100.113.5.197:9100,100.113.15.231:9100",
    "disk_encrypted": false,
    "enable_auto_topic": true
}
```


Status Code

Table 5-9 describes the status code of successful operations. For details about other status codes, see **Status Code**.

Table 5-9 Status code

Status Code	Description
200	Specified instance queried successfully.

5.3 Modifying an Instance

Function

This API is used to modify the name and description of an instance.

URI

PUT /v1.0/{project_id}/instances/{instance_id}

Table 5-10 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.
instance_id	String	Yes	Instance ID.

Request

Request parameters

Table 5-11 describes the parameters.

Table 5-11 Request parameters

Parameter	Type	Mandatory	Description
name	String	No	Instance name. An instance name consists of 4 to 64 characters including letters, digits, and hyphens (-) and must start with a letter.

Parameter	Type	Mandatory	Description
description	String	No	Description of an instance. It is a character string containing not more than 1024 characters. **NOTE** The backslash (\) and quotation mark (") are special characters for JSON packets. When using these characters in a parameter value, add the escape character (\) before these characters, for example, **** and **\"**.
maintain_begin	String	No	Time at which a maintenance time window starts. Format: HH:mm:ss • The start time and end time of the maintenance time window must indicate the time segment of a supported maintenance time window. For details about how to query the time segments of supported maintenance time windows, see **Querying Maintenance Time Windows**. • The start time must be set to 22:00:00, 02:00:00, 06:00:00, 10:00:00, 14:00:00, or 18:00:00. • Parameters **maintain_begin** and **maintain_end** must be set in pairs. If parameter **maintain_begin** is left blank, parameter **maintain_end** is also left blank. In this case, the system automatically sets the start time to 02:00:00.

Parameter	Type	Mandatory	Description
maintain_end	String	No	Time at which a maintenance time window ends. Format: HH:mm:ss • The start time and end time of the maintenance time window must indicate the time segment of a supported maintenance time window. For details about how to query the time segments of supported maintenance time windows, see **Querying Maintenance Time Windows**. • The end time is four hours later than the start time. For example, if the start time is 22:00:00, the end time is 02:00:00. • Parameters **maintain_begin** and **maintain_end** must be set in pairs. If parameter **maintain_end** is left blank, parameter **maintain_start** is also left blank. In this case, the system automatically sets the end time to 06:00:00.
security_group_id	String	No	Security group ID.
retention_policy	String	No	Capacity threshold policy. Options: • **produce_reject**: New messages cannot be created. • **time_base**: The earliest messages are deleted.
enterprise_project_id	String	No	Enterprise project.

Example request

Example 1:

```
{
   "name": "dms002",
   "description": "instance description"
}
```

Example 2:

```
{
    "name": "dms002",
```

```
    "description": "instance description",
    "maintain_begin":"02:00:00",
    "maintain_end":"06:00:00"
}
```

Response

Response parameters

None.

Example response

None.

Status Code

Table 5-12 describes the status code of successful operations. For details about other status codes, see Status Code.

Table 5-12 Status code

Status Code	Description
204	The instance is modified successfully.

5.4 Deleting an Instance

Function

This API is used to delete an instance to release all the resources occupied by it.

URI

DELETE /v1.0/{project_id}/instances/{instance_id}

Table 5-13 describes the parameters.

Table 5-13 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.
instance_id	String	Yes	Instance ID.

Request

Request parameters

None.

Example request

None.

Response

Response parameters

None.

Example response

None.

Status Code

Table 5-14 describes the status code of successful operations. For details about other status codes, see Status Code.

Table 5-14 Status code

Status Code	Description
204	The instance is deleted successfully.

5.5 Restarting or Deleting Instances in Batches

Function

This API is used to restart or delete instances in batches.

When an instance is being restarted, message retrieval and creation requests of the client will be rejected.

Deleting an instance will delete the data in the instance without any backup. Exercise caution when performing this operation.

URI

POST /v1.0/{project_id}/instances/action

Table 5-15 describes the parameters.

Table 5-15 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.

Request

Request parameters

Table 5-16 describes the parameters.

Table 5-16 Request parameters

Parameter	Type	Mandatory	Description
action	String	Yes	Operation to be performed on instances. The value of this parameter can be **restart** or **delete**.
instances	Array	No	List of instance IDs.
allFailure	String	No	When set to **kafka**, indicates all Kafka instances that fail to be created are to be deleted.

Example request

Restarting instances in batches:

```
{
   "action" : "restart",
   "instances" : ["54602a9d-5e22-4239-9123-77e350df4a34", "7166cdea-dbad-4d79-9610-7163e6f8b640"]
}
```

Deleting instances in batches:

```
{
   "action" : "delete",
   "instances" : ["54602a9d-5e22-4239-9123-77e350df4a34", "7166cdea-dbad-4d79-9610-7163e6f8b640"]
}
```

Deleting all instances that fail to be created:

```
{
   "action" : "delete",
   "allFailure" : "kafka"
}
```

Response

Response parameters

When **action** is set to **delete**, **allFailure** is set to **kafka**, and an empty response is returned, the instances are deleted successfully. Table 5-17 describes the parameters.

Table 5-17 Response parameters

Parameter	Type	Description
results	Array	Result of instance modification.

Table 5-18 results parameter description

Parameter	Type	Description
instance	String	Instance ID.
result	String	Instance modification result. Options: **success** and **failed**.

Example response

```
{
   "results": [
      {
         "result": "success",
         "instance": "afc90a2a-a02c-4cba-94d5-58dfa9ad1e0d"
      },
      {
         "result": "success",
         "instance": "67fc5f8d-3986-4f02-bb75-4075a23112de"
      }
   ]
}
```

Status Code

Table 5-19 describes the status code of successful operations. For details about other status codes, see **Status Code**.

Table 5-19 Status code

Status Code	Description
200	The instances are restarted or deleted successfully.
204	Successfully deleting an instance failed to be created.

5.6 Querying All Instances

Function

This API is used to query the instances of a tenant by set conditions.

URI

GET /v1.0/{project_id}/instances?engine={engine}&name={name}&status={status}&id={id}&includeFailure={includeFailure}&exactMatchName={exactMatchName}&enterprise_project_id={enterprise_project_id}

Table 5-20 describes the parameters.

Table 5-20 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.
engine	String	No	Message engine. Value: **kafka**
name	String	No	Instance name.
id	String	No	Instance ID.
status	String	No	Instance status. For details, see **Instance Status**.
includeFailure	String	No	Whether to return the number of instances that fail to be created. If the value is **true**, the number of instances that failed to be created is returned. If the value is not **true**, the number is not returned.
exactMatchName	String	No	Whether to search for the instance that precisely matches a specified instance name. The default value is **false**, indicating that a fuzzy search is performed based on a specified instance name. If the value is **true**, the instance that precisely matches a specified instance name is queried.
enterprise_project_id	String	No	Enterprise project ID.

Example

```
GET /v1.0/bd6b78e2ff9e4e47bc260803ddcc7a21/instances?
start=1&limit=10&name=&status=&id=&includeFailure=true&exactMatchName=false
```

Request

Request parameters

None.

Example request

None.

Response

Response parameters

Table 5-21 describes the parameters.

Table 5-21 Response parameters

Parameter	Type	Description
instances	Array	Instance details.
instance_num	Integer	Number of instances.

Table 5-22 instance parameter description

Parameter	Type	Description
name	String	Instance name.
engine	String	Message engine.
engine_version	String	Engine version.
specification	String	Instance specification.
storage_space	Integer	Message storage space. Unit: GB
partition_num	String	Maximum number of topics in a Kafka instance.
used_storage_space	Integer	Used message storage space. Unit: GB
connect_address	String	Instance IP address.
port	Integer	Instance port.
status	String	Instance status. For details, see **Instance Status**.
instance_id	String	Instance ID.
resource_spec_code	String	Resource specifications identifier. • **dms.instance.kafka.cluster.c3.mini**: Kafka instance, 100 MB/s reference bandwidth • **dms.instance.kafka.cluster.c3.small.2**: Kafka instance, 300 MB/s reference bandwidth • **dms.instance.kafka.cluster.c3.middle.2**: Kafka instance, 600 MB reference bandwidth • **dms.instance.kafka.cluster.c3.high.2**: Kafka instance, 1200 MB reference bandwidth
charging_mode	Integer	Billing mode.
vpc_id	String	VPC ID.

Parameter	Type	Description
vpc_name	String	VPC name.
created_at	String	Time when the instance is created. The time is in the format of timestamp, that is, the offset milliseconds from 1970-01-01 00:00:00 UTC to the specified time.
user_id	String	User ID.
user_name	String	Username.
maintain_begin	String	Time at which a maintenance time window starts. Format: HH:mm:ss
maintain_end	String	Time at which the maintenance time window ends. Format: HH:mm:ss
management_connect_address	String	Connection address of Kafka Manager of the Kafka instance.
ssl_enable	Boolean	Whether security authentication is enabled. ● **true**: enable ● **false**: disable
enterprise_project_id	String	Enterprise project ID.
is_logical_volume	Boolean	Distinguishes old instances from new instances during instance capacity expansion. ● **true**: New instance, which allows dynamic disk capacity expansion without restarting the instance. ● **false**: Old instance.
extend_times	Integer	Number of disk expansion times. If it exceeds 20, the disk cannot be expanded.
enable_auto_topic	Boolean	Whether automatic topic creation is enabled. ● **true**: enabled ● **false**: disabled
type	String	Instance type. Value: **cluster**.
product_id	String	Product ID.
security_group_id	String	Security group ID.
security_group_name	String	Security group name.

Parameter	Type	Description
subnet_id	String	Subnet ID.
available_zones	Array	ID of the AZ to which the instance node belongs. The AZ ID is returned.
total_storage_space	Integer	Message storage space. Unit: GB
storage_resource_id	String	Storage resource ID.
storage_spec_code	String	I/O specification.
service_type	String	Service type.
storage_type	String	Storage type.
retention_policy	String	Message retention policy.
kafka_manager_user	String	Username for logging in to Kafka Manager.
enable_log_collection	Boolean	Whether log collection is enabled.
cross_vpc_info	String	Cross-VPC access information.
ipv6_enable	Boolean	Whether IPv6 is enabled.
ipv6_connect_addresses	Array of strings	IPv6 connection address.
rest_enable	Boolean	Whether the Kafka REST function is enabled.
rest_connect_address	String	Kafka REST address.
message_query_inst_enable	Boolean	Whether message query is enabled.
vpc_client_plain	Boolean	Whether intra-VPC plaintext access is enabled.
support_features	String	List of features supported by the Kafka instance.
trace_enable	Boolean	Whether message tracing is enabled.
pod_connect_address	String	Connection address on the tenant side.
disk_encrypted	Boolean	Whether disk encryption is enabled.
kafka_private_connect_address	String	Private connection address of a Kafka instance.

Parameter	Type	Description
ces_version	String	Cloud Eye version.
tags	Array<Object>	List of tags.

Table 5-23 tags

Parameter	Type	Description
key	String	Tag key.
value	String	Tag value.

Example response


```
{
   "instances": [
      {
         "name": "kafka-l00230526",
         "engine": "kafka",
         "port": 9092,
         "status": "RUNNING",
         "type": "cluster",
         "specification": "100MB",
         "engine_version": "XXX",
         "connect_address": "192.168.1.116,192.168.1.152,192.168.1.78",
         "instance_id": "ef84dd5f-3ece-4336-8c99-987defd62e3a",
         "resource_spec_code": "dms.instance.kafka.cluster.c3.mini",
         "charging_mode": 1,
         "vpc_id": "2477879f-aebf-496f-a08a-67812885ce9b",
         "vpc_name": "vpc-y00502467",
         "created_at": "1568797295209",
         "product_id": "00300-30308-0--0",
         "security_group_id": "008a08e2-10cc-4d9b-90ab-3f3b8f6c3333",
         "security_group_name": "z00417080-cce-node-na7j",
         "subnet_id": "5ca08fb7-7522-4d95-9fa5-ff6b3592a29d",
         "available_zones": [
            "ae04cf9d61544df3806a3feeb401b204"
         ],
         "user_id": "2b4af4428ec840dfa1f0f1a32e965567",
         "user_name": "laiyh",
         "kafka_manager_user": "root",
         "maintain_begin": "22:00",
         "maintain_end": "02:00",
         "storage_space": 492,
         "total_storage_space": 600,
         "used_storage_space": 25,
         "partition_num": "300",
         "ssl_enable": false,
         "management_connect_address": "https://192.168.1.116:9999",
         "storage_resource_id": "81982562-ce8b-490a-95fa-2b225c292271",
         "storage_spec_code": "dms.physical.storage.ultra",
         "service_type": "advanced",
         "storage_type": "hec",
         "enterprise_project_id": "0",
         "is_logical_volume": true,
         "extend_times": 0,
         "retention_policy": "produce_reject",
         "ipv6_enable": false,
```



```
            "ipv6_connect_addresses": [],
            "rest_enable": false,
            "rest_connect_address": "",
            "message_query_inst_enable": true,
            "vpc_client_plain": false,
            "support_features":
"feature.physerver.kafka.topic.accesspolicy,message_trace_enable,features.pod.token.access,feature.physerver.
kafka.pulbic.dynamic,feature.physerver.kafka.user.manager",
            "trace_enable": false,
            "agent_enable": false,
            "pod_connect_address": "100.113.16.105:9100,100.113.5.197:9100,100.113.15.231:9100",
            "disk_encrypted": false,
            "enable_auto_topic": true
        }
    ],
    "instance_num": 1
}
```


Status Code

Table 5-24 describes the status code of successful operations. For details about other status codes, see Status Code.

Table 5-24 Status code

Status Code	Description
200	All instances are queried successfully.

5.7 Creating a Topic in a Kafka Instance

Function

This API is used to create a topic in a Kafka instance.

URI

POST /v1.0/{project_id}/instances/{instance_id}/topics

Table 5-25 describes the parameters.

Table 5-25 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.
instance_id	String	Yes	Instance ID.

Request

Request parameters

Table 5-26 describes the parameter.

Table 5-26 Request parameters

Parameter	Type	Mandatory	Description
id	String	Yes	Topic name. A topic name consists of 4 to 64 characters, starts with a letter, and contains only letters, hyphens (-), underscores (_), and digits.
partition	Integer	No	Number of topic partitions, which is used to set the number of concurrently consumed messages. Value range: 1–100. Default value: **3**.
replication	Integer	No	Number of replicas, which is configured to ensure data reliability. Value range: 1–3. Default value: **3**.
sync_replication	Boolean	No	Whether to enable synchronous replication. After this function is enabled, the **acks** parameter on the producer client must be set to **–1**. Otherwise, this parameter does not take effect. By default, synchronous replication is disabled.
retention_time	Integer	No	Retention period of a message. Its default value is **72**. Value range: 1–168. Default value: **72**. Unit: hour.
sync_message_flush	Boolean	No	Whether to enable synchronous flushing. Default value: **false**. Synchronous flushing compromises performance.

Example request

```
{
  "id" : "haha",
  "partition" : 3,
  "replication" : 3,
  "sync_replication " : true,
  "retention_time" : 10,
  "sync_message_flush" : true
}
```

Response

Response parameters

Table 5-27 describes the parameter.

Table 5-27 Response parameters

Parameter	Type	Description
id	String	Topic name.

Example response

```
{
"id": "haha"
}
```

Status Code

Table 5-28 describes the status code of successful operations. For details about other status codes, see **Status Code**.

Table 5-28 Status code

Status Code	Description
200	The topic is created successfully.

5.8 Querying a Topic in a Kafka Instance

Function

This API is used to query details about a topic in a Kafka instance.

URI

GET /v1.0/{project_id}/instances/{instance_id}/topics

Table 5-29 describes the parameter.

Table 5-29 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.

Parameter	Type	Mandatory	Description
instance_id	String	Yes	Instance ID.

Request

Request parameters

None.

Example request

None.

Response

Response parameters

Table 5-30 describes the response parameter.

Table 5-30 Response parameter

Parameter	Type	Description
total	Integer	Total number of topics.
size	Integer	Maximum number of records to be displayed on a page.
remain_partitions	Integer	Number of remaining partitions.
max_partitions	Integer	Total number of partitions.
topics	Array	List of topics.

Table 5-31 Parameter description

Parameter	Type	Description
policiesOnly	Boolean	Whether this policy is the default policy.
id	String	Topic name.
replication	Integer	Number of replicas, which is configured to ensure data reliability.
partition	Integer	Number of topic partitions, which is used to set the number of concurrently consumed messages.

Parameter	Type	Description
retention_time	Integer	Retention period of a message.
sync_replication	Boolean	Whether to enable synchronous replication. After this function is enabled, the **acks** parameter on the producer client must be set to **–1**. Otherwise, this parameter does not take effect. By default, synchronous replication is disabled.
sync_message_flush	Boolean	Whether synchronous flushing is enabled. Synchronous flushing compromises performance.
external_configs	Object	Extended configuration.
topic_type	Integer	Topic type.

Example response

```
{
 "count": 1,
 "topics": [
  {
   "id": "topic-test",
   "replication": 3,
   "partition": 4,
   "retention_time": 72,
   "sync_replication": "false",
   "sync_message_flush": "false"
  }
 ]
}
```

Status Code

Table 5-32 describes the status code of successful operations. For details about other status codes, see Status Code.

Table 5-32 Status code

Status Code	Description
200	The information is queried successfully.

5.9 Deleting Topics in a Kafka Instance in Batches

Function

This API is used to delete topics in a Kafka instance in batches.

URI

POST /v1.0/{project_id}/instances/{instance_id}/topics/delete

Table 5-33 describes the parameter.

Table 5-33 Parameters

Parameter	Type	Mandatory	Description
project_id	String	Yes	Project ID.
instance_id	String	Yes	Instance ID.

Request

Request parameters

Table 5-34 describes the parameter.

Table 5-34 Request parameter

Parameter	Type	Mandatory	Description
topics	Array	Yes	List of topics to delete.

Example request

```
{
 "topics" : ["hah", "aabb"]
}
```

Response

Response parameters

Table 5-35 describes the parameter.

Table 5-35 Response parameters

Parameter	Type	Description
topics	Array	List of topics.

Table 5-36 topics parameter description

Parameter	Type	Description
id	String	Topic name.
success	Boolean	Whether the topic is deleted.

Example response

```
{
  "topics" : [{
      "id" : "haha",
      "success" : true
    }, {
      "id" : "aabb",
      "success" : true
    }
  ]
}
```

Status Code

Table 5-37 describes the status code of successful operations. For details about other status codes, see **Status Code**.

Table 5-37 Status code

Status Code	Description
200	The topics are successfully deleted.

6 Other APIs

6.1 Querying AZ Information

Function

This API is used to query the AZ ID.

URI

GET /v1.0/availableZones

Request

Request parameters

None.

Example request

None.

Response

Response parameters

Table 6-1 and Table 6-2 describe the parameters.

Table 6-1 Response parameters

Parameter	Type	Description
region_id	String	Region ID.
available_zones	Array	Array of AZs. For details, see Table 6-2.

Table 6-2 available_zones parameter description

Parameter	Type	Description
soldOut	Boolean	Whether resources are sold out.
id	String	AZ ID.
code	String	AZ code.
name	String	AZ name.
port	String	AZ port.
resource_availability	String	Whether the AZ has available resources. • **true**: The AZ has available resources. • **false**: Resources of the AZ have been sold out.

Example response

```
{
   regionId: "XXXX",
   available_zones:[
      {
         "id":"1d7b939b382c4c3bb3481a8ca10da768",
         "name":"az10.dc1",
         "code":"az10.dc1",
         "port":"8002",
         "resource_availability": "true"
      },
      {
         "id":"1d7b939b382c4c3bb3481a8ca10da769",
         "name":"az10.dc2",
         "code":"az10.dc2",
         "port":"8002",
         "resource_availability": "true"
      }
   ]
}
```

Status Code

Table 6-3 describes the status code of successful operations. For details about other status codes, see Status Code.

Table 6-3 Status code

Status Code	Description
200	The AZ information is successfully queried.

6.2 Querying Product Specifications

Function

This API is used to query the product specifications to configure the product ID.

URI

GET /v1.0/products?engine={engine}

Table 6-4 describes the parameter.

Table 6-4 Parameters

Parameter	Type	Mandatory	Description
engine	String	Yes	Message engine.

Request

Request parameters

None.

Example request

None.

Response

Response parameters

Table 6-5 describes the response parameters.

Table 6-5 Response parameters

Parameter	Type	Description
name	String	Message engine, which is **kafka**.
version	String	Version of the message engine. Currently, only 1.1.0 and 2.3.0 are supported.
values	Array	List of product specifications. For details, see Table 6-6.

Table 6-6 values parameter description

Parameter	Type	Description
detail	Array	Specification details. For details, see **Table 6-7**.
name	String	Instance type.
unavailable_zones	Array	AZs where resources are sold out.
available_zones	Array	AZs where there are available resources.

Table 6-7 detail parameter description

Parameter	Type	Description
tps	String	Maximum number of messages per unit time.
storage	String	Message storage space.
partition_num	String	Maximum number of topics in a Kafka instance.
product_id	String	Product ID.
spec_code	String	Specification ID.
io	Array	I/O information. For details, see **Table 6-8**.
bandwidth	String	Bandwidth of a Kafka instance.
available_zones	Array	AZs where there are available resources.
ecs_flavor_id	String	Flavors of the corresponding ECS.
arch_type	String	Instance architecture type. Currently, only x86 is supported.

Table 6-8 io parameter description

Parameter	Type	Description
io_type	String	I/O type.
storage_spec_code	String	I/O specification.
available_zones	Array	List of AZs where there are available I/O resources.
unavailable_zones	Array of strings	List of AZs where I/O resources are sold out.
volume_type	String	Disk type.

Example response


```
{
   "Hourly": [{
      "name": "kafka",
      "version": "2.3.0",
      "values": [{
         "detail": [{
            "tps": "50000",
            "storage": "600",
            "partition_num": "300",
            "product_id": "00300-30308-0--0",
            "spec_code": "dms.instance.kafka.cluster.c3.mini",
            "io": [{
               "io_type": "high",
               "storage_spec_code": "dms.physical.storage.high",
               "available_zones": ["XXX",
               "XXX"],
               "volume_type": "SAS"
            },
            {
               "io_type": "ultra",
               "storage_spec_code": "dms.physical.storage.ultra",
               "available_zones": ["XXX",
               "XXX"],
               "volume_type": "SSD"
            }],
            "bandwidth": "100MB",
            "unavailable_zones": ["XXX"],
            "available_zones": ["XXX"],
            "ecs_flavor_id": "c4.large.2",
            "arch_type": "X86"
         },
         {
            "tps": "100000",
            "storage": "1200",
            "partition_num": "900",
            "product_id": "00300-30310-0--0",
            "spec_code": "dms.instance.kafka.cluster.c3.small.2",
            "io": [{
               "io_type": "high",
               "storage_spec_code": "dms.physical.storage.high",
               "available_zones": ["XXX",
               "XXX"],
               "volume_type": "SAS"
            },
            {
               "io_type": "ultra",
               "storage_spec_code": "dms.physical.storage.ultra",
               "available_zones": ["XXX",
               "XXX"],
               "volume_type": "SSD"
            }],
            "bandwidth": "300MB",
            "unavailable_zones": ["XXX"],
            "available_zones": ["XXX"],
            "ecs_flavor_id": "c4.xlarge.2",
            "arch_type": "X86"
         },
         {
            "tps": "200000",
            "storage": "2400",
            "partition_num": "1800",
            "product_id": "00300-30312-0--0",
            "spec_code": "dms.instance.kafka.cluster.c3.middle.2",
            "io": [{
               "io_type": "ultra",
```



```
                "storage_spec_code": "dms.physical.storage.ultra",
                "available_zones": ["XXX",
                "XXX"],
                "volume_type": "SSD"
            }],
            "bandwidth": "600MB",
            "unavailable_zones": ["XXX"],
            "available_zones": ["XXX"],
            "ecs_flavor_id": "c4.2xlarge.2",
            "arch_type": "X86"
        },
        {
            "tps": "300000",
            "storage": "4800",
            "partition_num": "1800",
            "product_id": "00300-30314-0--0",
            "spec_code": "dms.instance.kafka.cluster.c3.high.2",
            "io": [{
                "io_type": "ultra",
                "storage_spec_code": "dms.physical.storage.ultra",
                "available_zones": ["XXX",
                "XXX"],
                "volume_type": "SSD"
            }],
            "bandwidth": "1200MB",
            "unavailable_zones": ["XXX"],
            "available_zones": ["XXX"],
            "ecs_flavor_id": "c4.2xlarge.2",
            "arch_type": "X86"
        }],
        "name": "cluster",
        "unavailable_zones": ["XXX"],
        "available_zones": ["XXX"]
    }]
  }]
}
```


Status Code

Table 6-9 describes the status code of successful operations. For details about other status codes, see **Status Code**.

Table 6-9 Status code

Status Code	Description
200	Product specifications queried successfully.

6.3 Querying Maintenance Time Windows

Function

This API is used to query the start and end time of the maintenance window.

URI

GET /v1.0/instances/maintain-windows

Request

Request parameters

None.

Example request

None.

Response

Response parameters

Table 6-10 and Table 6-11 describe the response parameters.

Table 6-10 Response parameters

Parameter	Type	Description
maintain_windows	Array	List of supported maintenance time windows.

Table 6-11 maintain_windows parameter description

Parameter	Type	Description
seq	Integer	Sequence number.
begin	String	Start time of the maintenance time window.
end	String	End time of the maintenance time window.
default	Boolean	Whether a maintenance time window is set to the default time segment.

Example response

```
{
   "maintain_windows": [{
      "default": false,
      "seq": 1,
      "begin": "22:00:00",
      "end": "02:00:00"
   },
   {
      "default": true,
      "seq": 2,
      "begin": "02:00:00",
      "end": "06:00:00"
   },
   {
      "default": false,
      "seq": 3,
```

```
        "begin": "06:00:00",
        "end": "10:00:00"
    },
    {
        "default": false,
        "seq": 4,
        "begin": "10:00:00",
        "end": "14:00:00"
    },
    {
        "default": false,
        "seq": 5,
        "begin": "14:00:00",
        "end": "18:00:00"
    },
    {
        "default": false,
        "seq": 6,
        "begin": "18:00:00",
        "end": "22:00:00"
    }]
}
```


Status Code

Table 6-12 describes the status code of successful operations. For details about other status codes, see Status Code.

Table 6-12 Status code

Status Code	Description
200	The maintenance time windows are queried successfully.

7 Permissions Policies and Supported Actions

This chapter describes fine-grained permissions management for your DMS for Kafka instances. If your account does not require individual IAM users, you can skip this chapter.

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

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

NOTE

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

Your account has all the permissions required to call all APIs, but IAM users under your account must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user wants to query Kafka instances using an API, the user must have been granted permissions that allow the **dms:instance:create** action.

Supported Actions

DMS for Kafka 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. Operations 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.
- API: REST APIs that can be called by a user who has been granted specific permissions.

- Action: Specific operations that are allowed or denied.
- IAM project or enterprise projects: type of projects for which an action will take effect. Policies that contain actions for both IAM and enterprise projects can be used and take effect for both IAM and Enterprise Management. Policies that only contain actions for IAM projects can be used and only take effect for IAM.

DMS for Kafka supports the following actions that can be defined in custom policies. Permissions must be obtained before calling DMS APIs. For details on how to obtain permissions, visit the Identity and Access Management help center.

Table 7-1 DMS for Kafka actions

Permissions	APIs	Actions	IAM Project	Enterprise Project
Creating an instance	POST /v1.0/{project_id}/instances	dms:instance:create	√	√
Viewing instance details	GET /v1.0/{project_id}/instances/{instance_id}	dms:instance:get	√	√
Modifying an instance	PUT /v1.0/{project_id}/instances/{instance_id}	dms:instance:modify	√	√
Deleting an instance	DELETE /v1.0/{project_id}/instances/{instance_id}	dms:instance:delete	√	√
Modifying instance specifications	POST /v1.0/{project_id}/instances/{instance_id}/extend	dms:instance:scale	√	√
Querying all instances	GET /v1.0/{project_id}/instances	dms:instance:list	√	√
Restarting an instance	POST /v1.0/{project_id}/instances/action	dms:instance:modifyStatus	√	√
Modifying Kafka Manager or Kafka instance password	This operation is supported only by using the console and not by calling APIs.	dms:instance:modifyAuthInfo	√	√

Permissions	APIs	Actions	IAM Project	Enterprise Project
Resetting Kafka Manager or Kafka instance password	This operation is supported only by using the console and not by calling APIs.	dms:instance:resetAuthInfo	√	√
Querying one or more background tasks that have not been deleted	This operation is supported only by using the console and not by calling APIs.	dms:instance:getBackgroundTask	√	√
Deleting a background task	This operation is supported only by using the console and not by calling APIs.	dms:instance:deleteBackgroundTask	√	√

8 Appendix

8.1 Status Code

Table 8-1 lists status codes.

Table 8-1 Status codes

Status Code	Name	Description
100	Continue	The server has received the initial part of the request and the client should continue to send the remaining part.
101	Switching Protocols	The requester has asked the server to switch protocols and the server has agreed to do so. The target protocol must be more advanced than the source protocol. For example, the current HTTP protocol is switched to a later version of HTTP.
200	OK	Request sent successfully.
201	Created	The request has been fulfilled, resulting in the creation of a new resource.
202	Accepted	The request has been accepted for processing, but the processing has not been completed.
203	Non-Authoritative Information	The request has been fulfilled.
204	NoContent	The server has successfully processed the request, but is not returning any response body. The status code is returned in response to an HTTP OPTIONS request.

Status Code	Name	Description
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed a part of the GET request.
300	Multiple Choices	There are multiple options for the requested resource. For example, this code could be used to present a list of resource characteristics and addresses from which the client such as a browser may choose.
301	Moved Permanently	This and all future requests have been permanently moved to the given URI indicated in this response.
302	Found	The requested resource was temporarily moved.
303	See Other	The response to the request can be found under another URI using a GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, it does not return any resources.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	This HTTP status code is no longer used.
400	BadRequest	Invalid request. The client should modify the request instead of re-initiating it.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	Reserved for future use.
403	Forbidden	The server has received the request and understood it, but the server is refusing to respond to it. The client should modify the request instead of re-initiating it.
404	NotFound	The requested resource cannot be found. The client should modify the request instead of re-initiating it.

Status Code	Name	Description
405	MethodNotAllowed	A request method is not supported for the requested resource. The client should modify the request instead of re-initiating it.
406	Not Acceptable	The server cannot fulfill the request based on the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The server timed out when waiting for the request. The client may re-initiate the request without any modification at any time.
409	Conflict	The request cannot be processed due to a conflict, such as an edit conflict between multiple simultaneous updates or the resource that the client attempts to create already exits.
410	Gone	The requested resource has been deleted permanently and will not be available again.
411	Length Required	The server refused to process the request because the request does not specify the length of its content.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The server refuses to process a request because the request is too large. The server may close the connection to prevent the client from continuing the request. If the server cannot process the request temporarily, the response will contain a Retry-After field.
414	Request-URI Too Large	The URI provided was too long for the server to process.
415	Unsupported Media Type	The server does not support the media type in the request.
416	Requested range not satisfiable	The requested range is invalid.
417	Expectation Failed	The server fails to meet the requirements of the Expect request-header field.

Status Code	Name	Description
422	UnprocessableEntity	The request is well-formed but is unable to be processed due to semantic errors.
429	TooManyRequests	The client has sent more requests than its rate limit is allowed within a given amount of time, or the server has received more requests than it is able to process within a given amount of time. In this case, the client should re-initiate requests after the time specified in the Retry-After header of the response expires.
500	InternalServerError	The server is able to receive the request but it could not understand the request.
501	Not Implemented	The server does not support the requested function.
502	Bad Gateway	The server was acting as a gateway or proxy and received an invalid request from a remote server.
503	ServiceUnavailable	The requested service is invalid. The client should modify the request instead of re-initiating it.
504	ServerTimeout	The request cannot be fulfilled within a given time. The response will reach the client only if the request carries the **timeout** parameter.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

8.2 Error Codes

Table 8-2 Error codes returned during instance creation

Error Code	Description
public.00.0001	Internal service error.
public.00.0002	Internal service error.
public.00.0003	Internal service error.
public.00.0004	Failed to create the VPC.
public.00.0005	Failed to create the security group.
public.00.0006	Failed to create the subnet.
public.00.0007	The subnet status is abnormal.

Error Code	Description
public.00.0008	Failed to create the ECS.
public.00.0009	Failed to create the ECS.
public.00.0010	Failed to create the ECS.
public.00.0011	Failed to bind an NIC to the ECS.
public.00.0013	Failed to start the ECS.
public.00.0014	Failed to start the ECS.
public.00.0015	Failed to stop the ECS.
public.00.0018	Failed to create the ECS because the ECS resource quota is insufficient.
public.00.0024	Failed to deploy the instance.
public.00.0025	Some nodes of the instance are faulty.
public.00.0042	Failed to connect to the instance.

Table 8-3 Error codes of Kafka instances

Status Code	Error Code	Description
400	111400002	The project ID format is invalid.
400	111400004	The request body is empty.
400	111400005	The message body is not in JSON format or contains invalid characters.
400	111400007	Unsupported type.
400	111400008	Unsupported version.
400	111400009	Invalid **product_id**.
400	111400010	Invalid instance name. The name must be 4 to 64 characters long. Only letters, digits, underscores (_), and hyphens (-) are allowed.
400	111400011	The instance description can contain a maximum of 1024 characters.

Status Code	Error Code	Description
400	11140001 2	The password does not meet the complexity requirements. An instance password must meet the following requirements: ● Must be a string consisting of 8 to 32 characters. ● Contain at least two of the following character types: – Lowercase letters – Uppercase letters – Digits – Special characters `~!@#$%^&*()-_=+|[{}];:',<.>/?
400	11140001 3	**vpc_id** in the request is empty.
400	11140001 4	**security_group_id** in the request is empty.
400	11140001 5	Invalid username. A username must be 4 to 64 characters long and consist of only letters, digits, and hyphens (-).
400	11140001 6	**subnet_id** in the request is empty.
400	11140001 8	This subnet must exist in the VPC.
400	11140001 9	The password does not meet the complexity requirements.
400	11140002 0	DHCP must be enabled for this subnet.
400	11140002 6	This operation is not allowed due to the instance status.
400	11140003 7	The **instanceParams** parameter in the request contains invalid characters or is not in JSON format.
400	11140003 8	The **periodNum** parameter in the request must be an integer.
400	11140004 2	The AZ does not exist.
400	11140004 6	This security group does not exist.
400	11140004 8	Invalid security group rules. Ensure that rules with the protocol being **ANY** are configured for both the inbound and outbound directions.

Status Code	Error Code	Description
400	111400060	This instance name already exists.
400	111400061	Invalid instance ID format.
400	111400062	Invalid request parameter. The status of an instance to be queried must be a value listed in **Table 8-4**.
400	111400063	Invalid configuration parameter {0}.
400	111400064	The **action** parameter in the request must be **delete** or **restart**.
400	111400065	The **instances** parameter in the request is empty.
400	111400066	Invalid configuration parameter {0}.
400	111400067	The **available_zones** parameter in the request must be an array that contains only one AZ ID.
400	111400068	The VPC does not exist.
400	111400080	Incorrect instance password.
400	111400081	Duplicate instance name.
400	111400099	The following instances in the Creating, Starting, Stopping, or Restarting state cannot be deleted: {}
400	111400100	The **instances** array can contain a maximum of 50 instance IDs.
400	111400101	The name of a Kafka topic must be 4 to 64 characters long and start with a letter. Only letters, digits, underscores (_), and hyphens (-) are allowed.
400	111400102	The number of partitions created for a Kafka topic must be within the range of 1–20.
400	111400103	The number of replicas created for a Kafka topic must be within the range of 1–20.
400	111400105	The message retention period of a Kafka topic must be within the range of 1–168.
400	111400106	Invalid maintenance time window.

Status Code	Error Code	Description
400	111400800	• Duplicate topic name. • Invalid request. You can create, delete, or query a topic only for a Kafka instance. • Invalid parameter. Error information varies by scenario.
401	111401001	Invalid token.
401	111401003	The token is missing.
401	111401004	The project ID and token do not match.
403	111403002	A tenant has the read-only permission and cannot perform operations on DMS.
403	111403003	This role does not have the permissions to perform this operation.
404	111404001	The requested URL does not exist.
404	111404022	This instance does not exist.
405	111405001	This request method is not allowed.
400	111400500	Invalid disk space.
400	111400125	Invalid **SPEC_CODE**.
400	111400124	The maximum number of disk expansion times has been reached.
500	111500000	Internal service error.
500	111500006	Internal service error.
500	111500017	Internal service error.
500	111500023	Internal service error.
500	111500024	Internal service error.

Status Code	Error Code	Description
500	111500025	Internal service error.
500	111500041	Internal service error.
500	111500051	Internal service error.
500	111500052	Internal service error.
500	111500053	Internal service error.
500	111500054	Internal service error.
500	111500070	Internal service error.
500	111500071	Internal service error.
500	111500094	Internal service error.
500	111500106	Internal service error.

8.3 Instance Status

Table 8-4 Instance status description

Status	Description
CREATING	The instance is being created.
CREATEFAILED	The instance fails to be created.
RUNNING	The instance is running properly. Instances in the **Running** state can provide services.
ERROR	The instance is not running properly.
RESTARTING	The instance is being restarted.
EXTENDING	The instance specifications are being changed.
EXTENDEDFAILED	The instance specifications failed to be changed.
DELETING	The instance is being deleted.

Status	Description
DELETED	The instance has been deleted.

8.4 Obtaining a Project ID

Obtaining a Project ID on the Console

A project ID is required for some URLs when an API is called. You can obtain a project ID on the console.

The following procedure describes how to obtain a project ID:

Step 1 Log in to the management console.

Step 2 Click the username in the upper right corner and choose **My Credential** from the drop-down list.

On the **My Credential** page, view the project IDs in the project list.

If there are multiple projects in one region, expand **Region** and view sub-project IDs in the **Project ID** column.

----End

8.5 Obtaining the Domain Name and Domain ID

The domain name and domain ID are required for some URLs when an API is called. To obtain the domain name and domain ID, perform the following operations:

1. Log in to the management console.
2. Click the username in the upper right corner and choose **My Credential** from the drop-down list.

   Viewing the domain name and domain ID

A Change History

Released On	Description
2022-08-12	This issue is the first official release.

