

Blockchain Service

API Reference

Issue	01
Date	2023-06-20



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2023. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 API Calling.....	1
1.3 Endpoints.....	1
1.4 Concepts.....	1
2 API Overview.....	3
3 Examples.....	6
3.1 Example 1: Creating a BCS Service.....	6
3.2 Example 2: Creating a Channel.....	7
3.3 Example 3: Inviting Tenants to Join a Consortium.....	7
4 Calling APIs.....	9
4.1 Making an API Request.....	9
4.2 Authentication.....	12
4.3 Response.....	13
5 APIs (Enhanced Hyperledger Fabric).....	15
5.1 BCS Management.....	15
5.1.1 Creating a BCS Service.....	15
5.1.2 Querying Creation Status of a BCS Service.....	26
5.1.3 Querying a BCS Service.....	31
5.1.4 Modifying a BCS Service.....	42
5.1.5 Creating Channels.....	48
5.1.6 Querying Channel Information.....	52
5.1.7 Adding Peers to a Channel.....	57
5.1.8 Removing Organizations from a Channel.....	61
5.1.9 Downloading Certificates.....	65
5.1.10 Downloading the SDK Configuration.....	69
5.1.11 Generating a User Certificate.....	74
5.1.12 Unfreezing a User Certificate.....	79
5.1.13 Freezing a User Certificate.....	83
5.1.14 Querying Quotas.....	87
5.1.15 Querying Flavors.....	92
5.1.16 Querying Peer Information.....	96

5.1.17 Querying Asynchronous Operation Results.....	101
5.1.18 Querying the BCS Service List.....	109
5.1.19 Deleting a BCS Service.....	113
5.1.20 Removing a Peer from a Channel.....	118
5.1.21 Deleting a Channel.....	122
5.2 BCS Consortium.....	126
5.2.1 Inviting Tenants to Join a Consortium.....	126
5.2.2 Processing an Invitation.....	131
5.2.3 Querying All Notifications.....	136
5.2.4 Listing Consortium Members.....	142
5.2.5 Deleting Invitation Information.....	147
5.2.6 Exiting a Consortium.....	152
5.3 BCS Monitoring.....	157
5.3.1 Querying BCS Alarms.....	157
5.3.2 Querying BCS Alarm Statistics.....	163
5.3.3 Querying BCS Monitoring Data.....	169
5.3.4 Listing Entity Monitoring Data of a BCS Service.....	175
5.3.5 Querying Entity Monitoring Details of a BCS Service.....	180
6 Permissions Policies and Supported Actions.....	187
6.1 Introduction.....	187
6.2 Actions Supported by Custom Policies.....	190
7 Appendix.....	194
7.1 Status Codes.....	194
7.2 Error Codes.....	197
7.3 Obtaining a Project ID.....	246
7.4 Obtaining the ID.....	247
8 Change History.....	248

1 Before You Start

1.1 Overview

Welcome to *Blockchain Service API Reference*. Blockchain Service (BSC) helps you quickly deploy, manage, and maintain a blockchain network in the cloud, reducing the threshold for using blockchain. With BCS, you can focus on the development and innovation of your own business and quickly migrate your business to blockchains.

In addition to the web console, you can also invoke APIs for managing your BCS O&M resources. This document describes how to use APIs to perform creation, query, and deletion operations on BCS. For details, see [APIs \(Enhanced Hyperledger Fabric\)](#).

If you plan to access a BCS service through an API, ensure that you are familiar with BCS concepts.

1.2 API Calling

BCS support 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](#).

Select an endpoint based on your service requirements.

1.4 Concepts

- Account

An account is created upon successful registration and 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 Identity and Access Management (IAM) users and grant them permissions for routine management.

- IAM user

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

An IAM user can view the account ID and user ID on the **My Credentials** page. The account name, username, and password will be required for API authentication.

- 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.

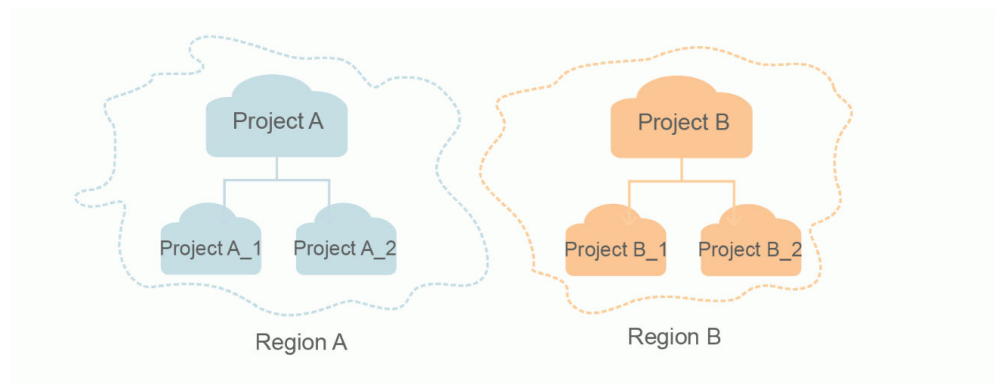
- AZ

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 support cross-AZ high-availability systems.

- Project

Default projects are defined to group and isolate resources (compute, storage, and network) across regions. You can grant users permissions to access resources in the region associated with a default 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.

Figure 1-1 Project isolation model



2 API Overview

You can call APIs to use the functions provided by BCS, including creating and deleting BCS services, and forming consortium blockchains. [Table 2-1](#) lists the APIs.

Table 2-1 API Overview

Category	API	Description
BCS Managem ent	Creating a BCS Service	Creates a BCS service.
	Querying Creation Status of a BCS Service	Queries the creation status of a specified BCS service.
	Querying Service Information	Queries the details of a specified BCS service.
	Modifying a BCS Service	Modifies the peers and organizations of a BCS service.
	Creating Channels	Creates channels.
	Querying Channel Information	Queries the channel details of a specified BCS service.
	Adding Peers to a Channel	Adds peers to a channel.
	Removing Organizations from a Channel	Removes organizations from a channel.
	Downloading Certificates	Downloads the certificates of a specified BCS service.
	Downloading the SDK Configuration	Downloads the SDK configuration of a specified BCS service.

Category	API	Description
	Generating a User Certificate	Generates a user certificate for a specific service.
	Unfreezing a User Certificate	Unfreezes a user certificate for a specific BCS service. It takes 0.5 minutes to 1 minute for the unfreezing to take effect.
	Freezing a User Certificate	Freezes a user certificate for a specific BCS service. It takes 0.5 minutes to 1 minute for the freezing to take effect.
	Querying Quotas	Queries the quotas of all BCS service resources under the specified project.
	Querying Flavors	Queries the basic information of all BCS services under the specified project.
	Querying Peer Information	Queries the peer details of a specified BCS service.
	Querying Asynchronous Operation Results	Queries the asynchronous operation results.
	Querying the BCS Service List	Queries the basic information of all BCS services under the specified project.
	Deleting a BCS Service	Deletes a BCS service.
	Removing a Peer from a Channel	Removes a peer from a channel.
	Deleting a Channel	Deletes a channel.
BCS Consortium	Inviting Tenants to Join a Consortium	Invites multiple tenants in batches to join a channel. Invitation notifications will be sent to the invited tenants.
	Processing an Invitation	Processes an invitation to join a consortium.
	Querying All Notifications	Queries all notifications.
	Listing Consortium Members	Lists all members in a consortium.
	Deleting Invitation Information	Cancels invitations in batches or delete invitation information about members who have quit, rejected to join, or dismissed.

Category	API	Description
	Exiting a Consortium	Exits a consortium.
BCS Monitoring	Querying BCS Alarms	Queries the monitoring data of a specified metric of a BCS service.
	Querying BCS Alarm Statistics	Queries the alarm statistics of a specified metric of a BCS service.
	Querying Monitoring Data of a BCS Service	Queries the monitoring data of a specified metric of a BCS service.
	Listing Entity Monitoring Data of a BCS Service	Queries the monitoring data list of entities of a BCS service.
	Querying Entity Monitoring Details of a BCS Service	Queries the monitoring details of entities of a specified BCS service.

3 Examples

3.1 Example 1: Creating a BCS Service

Scenario

This section describes how to create a BCS service by calling an API.

Related APIs of Other Services

None

Creating a BCS Service

- URI format
POST `/v2/{project_id}/blockchains`
- Example request
POST `https://{bcs_endpoint}/v2/{project_id}/blockchains`
Obtain the value of `{bcs_endpoint}` from [Regions and Endpoints](#).

Body:

```
{
  "name": "bcs-test",
  "fabric_version": "2.0",
  "enterprise_project_id": "0",
  "resource_password": "xxx",
  "create_new_cluster": false,
  "cce_cluster_info": {
    "cluster_id": "c11ccc2c-aab9-11ea-99da-0255ac100b03",
    "cluster_name": "byl-fastfabric"
  }
}
```

- Example response

```
{
  "blockchain_id": "b95133b2-b933-8fd5-3421-e3ce793a3e2f",
  "blockchain_name": "bcs-test"
}
```

3.2 Example 2: Creating a Channel

Scenario

This section describes how to create a channel by calling an API.

Related APIs of Other Services

None

Creating a Channel

- URI format
POST `/v2/{project_id}/blockchains/{blockchain_id}/channels`
- Example request
POST `https://{bcs_endpoint}/v2/{project_id}/blockchains/{blockchain_id}/channels`
Obtain the value of `{bcs_endpoint}` from [Regions and Endpoints](#).
Body:

```
{
  "channels": [
    {
      "channel_name": "channel001",
      "channel_description": "channel001"
    }
  ]
}
```
- Example response
If the operation is successful, the returned value is empty.

3.3 Example 3: Inviting Tenants to Join a Consortium

Scenario

This section describes how to invite tenants to join a consortium by calling an API.

Related APIs of Other Services

None

Inviting Tenants to Join a Consortium

- URI format
POST `/v2/{project_id}/members/invitations`
- Example request
POST `https://{bcs_endpoint}/v2/{project_id}/members/invitations`
Obtain the value of `{bcs_endpoint}` from [Regions and Endpoints](#).
Body:

```
{
  "bcs_id": "ba0fc16d-c752-9ac4-7a43-7a84443e0a06",
  "channel_name": "channel",
  "invitor_username": "paas_bcs_c00218797_02",
  "invited_userinfo": [
    {
      "invited_username": "paas_bcs_bwx705565_02"
    }
  ]
}
```

- Example response

If the operation is successful, the returned value is empty.

4 Calling APIs

4.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for [obtaining a user token through password authentication](#) 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 consists of the following parts:

{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](#).

For example, the endpoint of BCS in the **CN North-Beijing1** region is **bcs.cn-north-1.myhuaweicloud.com**.

- **resource-path**

Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.

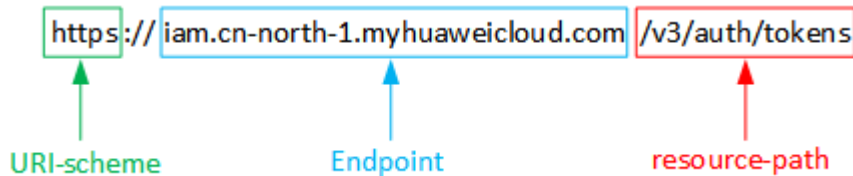
- **query-string**

Query parameter, which is optional. Query parameter, which is optional. Ensure that a question mark (?) is included before each 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.

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

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

Figure 4-1 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 HTTPS protocol defines the following request methods that can be used to send a request to the server:

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

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

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

Request Header

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

Common request header fields are as follows:

- **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, 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 through password authentication](#) does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

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

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 through password authentication](#), the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace **username**, **domainname**, ********* (login password), and **xxxxxxxxxxxxxxxxxxxx** (project ID) with the actual values. To learn how to obtain a project ID, see [Obtaining a Project ID](#).

NOTE

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

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

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

```
    "id": "xxxxxxxxxxxxxxxxxxxxx"
  }
}
```

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.

4.2 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. Use a public account as the account for issuing the AK/SK. Do not use a personal account.

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.

When calling an API to obtain a user token, you must set **auth.scope** in the request body to **project**.

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

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

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

AK/SK-based Authentication

NOTE

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

In AK/SK-based authentication, AK/SK is used to sign requests and the signature is then added to the request headers 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 signature SDK, see [API Signature Guide](#).

NOTICE

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

4.3 Response

Status Code

After sending a request, you will receive a response, including 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 response. For more information, see [Status Codes](#).

For example, if status code **201** is returned for calling the [Obtaining a User Token Through Password Authentication](#) API, the request is successful.

Response Header

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

Figure 4-2 shows the response header fields for the API used to [obtain a user token through password authentication](#). The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 4-2 Header fields of the response to the request for obtaining a user token

```
connection → keep-alive

content-type → application/json

date → Tue, 12 Feb 2019 06:52:13 GMT

server → Web Server

strict-transport-security → max-age=31536000; includeSubdomains;

transfer-encoding → chunked

via → proxy A

x-content-type-options → nosniff

x-download-options → noopen

x-frame-options → SAMEORIGIN

x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5

x-subject-token
→ MIIYXQYJKoZIhvcNAQcColIYTCCEGoCAQExDTALBgIghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOnsiZXhwaXJlc19hdCI6IjIwMTktMDItMTNUMC
fj3KJs6YgKnpVNRbW2eZ5eb78SZOkqjACgklqO1wi4JIGzrpd18LGXK5bdfq4lqHCYb8P4NaY0NYejcAgz/VeFYtLWT1GSO0zxKZmlQHq82HBqHdgIZO9fuEbL5dMhdavj+33wEI
xHRCE9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXl1jipPEGA270g1FruooL6jggIFkNPQuFSOU8+uSsttVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUbpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==

x-xss-protection → 1; mode=block;
```

Response Body

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

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

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

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

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

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

5 APIs (Enhanced Hyperledger Fabric)

5.1 BCS Management

5.1.1 Creating a BCS Service

Function

This API is used to create a pay-per-use BCS service.

URI

POST /v2/{project_id}/blockchains

Table 5-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Table 5-3 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	BCS service name. The name can contain 4 to 24 characters, including letters, digits, and hyphens (-). It cannot start with a hyphen (-).
version_type	Yes	Long	BCS versions. Options: 4 (Basic), 2 (Professional) If the service is created by an invitee, the value must be the same as that of the inviting party.
fabric_version	No	String	Fabric version. The value can only be 2.2 if the service was created by an inviting party or is a private blockchain. If the service was created by an invitee, the value must be the same as the version of the inviting party's blockchain. The CCE cluster used by Fabric v1.4 services can only be v1.15 or earlier. The default value is 2.2 .
blockchain_type	No	String	Blockchain type. The value can be union (consortium blockchain) or private (private blockchain). Default value: private . If the service is created by an invitee, the value must be the same as that of the inviting party.
consensus	No	String	Consensus algorithm used by the BCS service. For Fabric v1.4, the value can be SOLO or SFLIC (fast Byzantine fault tolerance). For Fabric v2.2, the value can be etcdraft or SFLIC (fast Byzantine fault tolerance). If the service is created by an invitee, the value must be the same as that of the inviting party.

Parameter	Mandatory	Type	Description
sign_algorithm	No	String	Security mechanism used by the BCS service. The value can be ECDSA or sm2 (OSCCA-published cryptographic algorithms).
enterprise_project_id	Yes	String	ID of the enterprise project that the BCS service belongs to.
volume_type	No	String	Type of the storage volume used by the CCE cluster. The value can be evs (EVS disk), nfs (SFS), or efs (SFS Turbo).
evs_disk_type	No	String	EVS disk type, which is required when volume_type is evs . The value can be SATA (common I/O), SAS (high I/O), or SSD (ultra-high I/O).
org_disk_size	No	Long	Storage capacity of the peer organization. Minimum value: 100 GB for the basic and professional editions.
database_type	No	String	Type of the database used by the BCS service, including the file database (<i>goleveldb</i> *) and NoSQL (couchdb). If couchdb is used, specify the couchdb_info field.
resource_password	Yes	String	Resource access and blockchain management password.
orderer_node_number	No	Long	Number of orderers. This parameter is not required when an invitee is creating a BCS service, but is required for a Fabric v2.2 service.
use_eip	No	Boolean	Whether to use the elastic IP address (EIP) of a CCE node.
bandwidth_size	No	Long	EIP bandwidth.
cluster_type	Yes	String	Cluster type, which can be a CCE cluster.

Parameter	Mandatory	Type	Description
create_new_cluster	Yes	Boolean	Indication of whether to create a cluster. If you use an existing cluster, set this parameter to the value of the cce_cluster_info field. If you create a cluster, set this parameter to the value of the cce_create_info field.
cce_cluster_info	No	CCEClusterInfo object	Information about an existing CCE cluster. The CCE cluster used by Fabric v1.4 services can only be v1.15 or earlier.
cce_create_info	No	CCECreateInfo object	Information of the CCE cluster being created.
ief_deploy_mode	No	Long	IEF cluster deployment mode. The value can be 0 (a node can be randomly allocated to an organization) or 1 (a peer organization is bound with a node). If this parameter is set to 1 , the peer_orgs parameter must be set. The organization name must be the same as the IEF node name.
ief_nodes_info	No	Array of IEFNode objects	IEF cluster nodes. This parameter is required when the BCS service is deployed using IEF.
peer_orgs	No	Array of OrgPeer objects	Peer organization list. If a peer organization is bound with a node, the organization must be named after the node. This parameter is required when the BCS service is deployed using IEF.
channels	No	Array of ChannelInfoV2 objects	Channel list.
couchdb_info	No	CouchDb object	CouchDB information.
turbo_info	No	TurboInfo object	Information about the SFS Turbo file system.
block_info	No	block_info object	Block generation configurations.

Parameter	Mandatory	Type	Description
kafka_create_info	No	KafkaCreateInfo object	Kafka instance creation information.
tc3_need	No	Boolean	Whether Trusted Computing Platform is enabled.
restful_api_support	No	Boolean	Whether RESTful APIs are enabled.
is_invitee	No	Boolean	Indication of whether the BCS service is created for an invitee. If yes, specify the invitor_infos field.
invitor_infos	No	InvitorInfos object	Inviting party information.

Table 5-4 CCEClusterInfo

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID.
cluster_name	Yes	String	Cluster name.
cluster_platform_type	Yes	String	Cluster CPU architecture type. The value can be VirtualMachine (x86) or ARM64 (Arm).

Table 5-5 CCECreateInfo

Parameter	Mandatory	Type	Description
node_num	Yes	Long	Number of CCE cluster nodes.
node_flavor	Yes	String	Flavor ID of a CCE cluster node (minimum 4 vCPUs and 8 GB memory).
cce_flavor	Yes	String	CCE cluster flavor.
init_node_password	Yes	String	Initial password of the node.
az	Yes	String	AZ.
cluster_platform_type	Yes	String	Cluster CPU architecture type. The value can be VirtualMachine (x86) or ARM64 (Arm).

Table 5-6 IEFNode

Parameter	Mandatory	Type	Description
id	Yes	String	Node ID.
status	Yes	String	Node state (ACTIVE).
public_ip_address	Yes	String	Node public IP address.

Table 5-7 OrgPeer

Parameter	Mandatory	Type	Description
name	Yes	String	Organization name.
node_count	Yes	Long	Number of peers in the organization.

Table 5-8 ChannelInfoV2

Parameter	Mandatory	Type	Description
name	Yes	String	Channel name.
org_names	Yes	Array of strings	Names of organizations in the channel.
description	No	String	Channel description.

Table 5-9 CouchDb

Parameter	Mandatory	Type	Description
user_name	Yes	String	CouchDB username.
password	Yes	String	CouchDB password.

Table 5-10 TurboInfo

Parameter	Mandatory	Type	Description
share_type	Yes	String	Sharing mode. The value is fixed at STANDARD .

Parameter	Mandatory	Type	Description
type	Yes	String	Type. The value is fixed to efs-ha .
available_zone	Yes	String	AZ. The value can be an empty string.
resource_spec_code	Yes	String	Specifications. The value is fixed at sfs.turbo.standard .

Table 5-11 block_info

Parameter	Mandatory	Type	Description
batch_timeout	No	Long	Block generation interval in seconds. The default interval is 2s.
max_message_count	No	Long	Number of transactions in a block. The default quantity is 500.
preferred_maxbytes	No	Long	Block size in MB. The default size is 2 MB.

Table 5-12 KafkaCreateInfo

Parameter	Mandatory	Type	Description
spec	Yes	String	Kafka instance specifications. The value can be mini (100 MB/s bandwidth), small (300 MB/s bandwidth), middle (600 MB/s bandwidth), or high (1200 MB/s bandwidth).
storage	Yes	Long	Storage space in GB. The maximum value is 9000 . The storage space is at least 300 for mini instances, 1200 for small instances, 2400 for middle instances, and 4800 for high instances.
az	Yes	String	AZs configured for the Kafka instance.

Table 5-13 InvitorInfos

Parameter	Mandatory	Type	Description
tenant_id	Yes	String	Tenant ID of the inviting party.
project_id	Yes	String	Project ID of the inviting party.
blockchain_id	Yes	String	BCS service ID of the inviting party.

Response Parameters

Status code: 200

Table 5-14 Response body parameters

Parameter	Type	Description
blockchain_id	String	BCS service instance ID.
blockchain_name	String	BCS service instance name.
operation_id	String	Operation ID.

Status code: 400

Table 5-15 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-16 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-17** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-18** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-19 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{
  "name" : "bcs-test",
  "fabric_version" : "2.2",
  "blockchain_type" : "union",
  "consensus" : "etcdraft",
  "version_type" : 4,
  "sign_algorithm" : "ECDSA",
  "enterprise_project_id" : "0",
  "volume_type" : "nfs",
  "evs_disk_type" : "SATA",
  "org_disk_size" : 100,
  "database_type" : "goleveldb",
  "resource_password" : "xxx",
  "orderer_node_number" : 3,
  "use_eip" : true,
  "bandwidth_size" : 5,
  "cluster_type" : "cce",
  "create_new_cluster" : false,
  "cce_cluster_info" : {
    "cluster_id" : "5adc2ce2-9712-11ea-9535-0255ac100b0d",
    "cluster_name" : "byl-1-15",
    "cluster_platform_type" : "ARM64"
  },
  "cce_create_info" : {
    "node_num" : 1,
    "node_flavor" : "Si1.xlarge.4.linux",
    "cce_flavor" : "cce.s1.small",
    "init_node_pwd" :
"JDYkUzdOOUR0JEdyajFQV0tSWXNQrjZBZlhiSE5MVmpWV204V3M5a1FMUXl4Z3dtOVNkb25xR0lIZWZaNUFReDVWeHNJa3gvd3JML2d1WjZyazNLQmNWczBpWVVF2R2gw",
    "az" : "cn-north-7a",
    "cluster_platform_type" : "ARM64"
  },
  "ief_deploy_mode" : 0,
  "ief_nodes_info" : [ {
    "status" : "ACTIVE",
    "public_ip_address" : "192.168.1.95",
    "id" : "5700372e-49ef-4f4c-b433-11d95ad7acea"
  }, {
    "status" : "ACTIVE",
    "public_ip_address" : "192.168.1.213",
    "id" : "2455e71a-d806-4827-9cf4-81e1b3764681"
  } ],
  "peer_orgs" : [ {
    "name" : "organization",
    "node_count" : 2
  } ],
  "channels" : [ {
    "name" : "channel",
```

```
"org_names" : [ "organization" ],
"description" : ""
}],
"block_info" : {
  "batch_timeout" : 2,
  "max_message_count" : 500,
  "preferred_maxbytes" : 2
},
"turbo_info" : {
  "share_type" : "STANDARD",
  "type" : "efs-ha",
  "available_zone" : "cn-north-7a",
  "resource_spec_code" : "sfs.turbo.standard"
}
}
```

Example Responses

Status code: 200

Ok

```
{
  "blockchain_id" : "12691561-a50d-5207-6041-7346c90d4499",
  "blockchain_name" : "bcs-api",
  "operation_id" : "BCSSVC01-03-1617158790255323683"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.2 Querying Creation Status of a BCS Service

Function

This API is used to query the creation status of a specified BCS service.

URI

GET /v2/{project_id}/blockchains/{blockchain_id}/status

Table 5-20 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID

Request Parameters

Table 5-21 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-22 Response body parameters

Parameter	Type	Description
bcs	Detail object	Creation status of the BCS service.
eip	Detail object	Creation status of the EIP.
sfs	Detail object	Creation status of the SFS file system.
obs	Detail object	Creation status of OBS storage.
kafka	Detail object	Creation status of the Kafka instance.
cce	ComCCE object	Creation status of CCE resources.

Table 5-23 ComCCE

Parameter	Type	Description
cluster	Detail object	Creation status of the CCE cluster.
network	Detail object	Creation status of the CCE network.
security_group	Detail object	Creation status of the CCE security group.

Table 5-24 Detail

Parameter	Type	Description
start_time	String	Start time.

Parameter	Type	Description
end_time	String	End time.
status	String	Status. Enumeration values: • waiting • deploying • finished • failed
detail	String	Details.

Status code: 400**Table 5-25** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-26** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-27 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-28** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-29** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "bcs" : {
    "start_time" : "2020-10-15T18:24:48.658236618+08:00",
    "end_time" : "2020-10-15T18:25:53.552512674+08:00",
    "status" : "finished",
    "detail" : ""
  },
  "eip" : {
    "start_time" : "2020-10-15T18:24:31.957780831+08:00",
    "end_time" : "2020-10-15T18:24:33.810699517+08:00",
    "status" : "finished",
    "detail" : ""
  },
  "sfs" : {
    "start_time" : "2020-10-15T18:24:31.957781285+08:00",
    "end_time" : "2020-10-15T18:24:48.611812564+08:00",
    "status" : "finished",
    "detail" : ""
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
}
```

```
"error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.3 Querying a BCS Service

Function

This API is used to query the details of a specified BCS service.

URI

GET /v2/{project_id}/blockchains/{blockchain_id}

Table 5-30 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID

Request Parameters

Table 5-31 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200**Table 5-32** Response body parameters

Parameter	Type	Description
basic_info	BasicInfo object	BCS service basic information.
channels	Array of ChannelInfo objects	Channel information.
peer_info	Array of PeerInfo objects	Peer organization information.
light_peer_info	Array of PeerInfo objects	light-node organization information.
orderer_info	PeerInfo object	Orderer organization information.
couch_db_info	CouchDBInfo object	Couch DB information.
dms_kafka_info	DmsKafkaInfo object	Creation status of the Kafka instance.
ief_info	IefInfo object	IEF edge cluster information.
sfs_info	SfsInfo object	SFS information.
agent_info	PeerInfo object	agent organization information.
restapi_info	PeerInfo object	restapi organization information.

Parameter	Type	Description
evs_pvc_info	map<string, object>	evs_pvc information.
tc3_taskserver_info	PeerInfo object	tc3_taskserver organization information.
obs_bucket_info	OBSInfo object	OBS bucket information.

Table 5-33 BasicInfo

Parameter	Type	Description
id	String	BCS service ID.
name	String	BCS service name. The name can contain 4 to 24 characters, including letters, digits, and hyphens (-). It cannot start with a hyphen (-).
version	String	BCS service version.
service_type	String	BCS service type. The value can be union or private .
purchase_type	String	Deployment type of the BCS service. The value can be onestep (quick deployment) or normal (normal deployment).
sign_algorithm	String	Security mechanism used by the BCS service. The value can be ECDSA or SM2 (OSCCA-published cryptographic algorithms).
consensus	String	Consensus mechanism of the BCS service. The value can be solo , sflic (FBFT), kafka , or etcdraft .
charging_mode	Long	Billing mode of the BCS service. The value can be 1 (pay-per-use).
version_type	Long	BCS service edition.
database_type	String	Type of the database used by the BCS service. The value can be goleveldb (file database) or couchdb (NoSQL).
cluster_id	String	ID of the cluster where the BCS service is deployed.
cluster_name	String	Name of the cluster where the BCS service is deployed.

Parameter	Type	Description
cluster_type	String	Type of the cluster where the BCS service is deployed. The value can be CCE (CCE cluster) or ief (IEF cluster).
cluster_az	String	Whether the BCS service is deployed in multiple AZs. The value can be yes (multiple AZs) or no (single AZ).
created_time	String	Time when the BCS service was created.
deploy_type	String	Available for consortium blockchains. The value can be create (inviting party) or invite (invitee).
order_fade_enabled	Boolean	Whether to enable data aging on orderers.
is_cross_region	Boolean	Whether the BCS service is deployed across regions.
is_support_rollback	Boolean	Whether rollback is supported when the BCS service fails to be upgraded.
is_support_restful	Boolean	Whether RESTful APIs are supported by the BCS service. The value can be true (supported) or false (not supported).
is_support_tc3	Boolean	Whether the TC3 add-on is supported. Options: true (supported) or false (not supported).
is_old_service	Boolean	Whether the BCS is an old service. The value can be true (old service) or false (new service).
old_service_version	String	Version of an old BCS service.
agent_portal_addrs	Array of strings	Agent addresses and port numbers on the user data plane of the BCS service.
status	String	BCS service status. The value can be Normal , Abnormal , EipAbnormal , Freeze (frozen), Hibernation (hibernating), or other values (unknown).
process_status	String	BCS service processing status. The value can be IsCreating , IsUpgrading , IsScaling , IsDeleting , or IsAdding (adding peers).
order_status	Long	Order status for a BCS service billed in the yearly/monthly mode. The value can be 0 (the order failed), 1 (the order is abnormal), or 2 (the order is normal).

Parameter	Type	Description
order_fade_cache	Long	Aging threshold of orderer data.
deploy_status	Integer	BCS service deployment status, which can be 0 (in progress), 1 (successful), 2 (failed), or 3 (finished).
block_info	block_info object	Block generation configurations.
cluster_platform_type	String	Cluster CPU architecture type. The value can be VirtualMachine (x86) or ARM64 (Arm).

Table 5-34 block_info

Parameter	Type	Description
batch_timeout	Long	Block generation interval in seconds. The default interval is 2s.
max_message_count	Long	Number of transactions in a block. The default quantity is 500.
preferred_maxbytes	Long	Block size in MB. The default size is 2 MB.

Table 5-35 ChannelInfo

Parameter	Type	Description
name	String	Channel name.
org_names	Array of strings	Names of organizations in the channel.
org_name_hash	Array of strings	Hash values of organization names in the channel.
peers	Map<String,Array<String>>	The key is the organization name and value is the peer array.

Table 5-36 PeerInfo

Parameter	Type	Description
name	String	Organization name.
node_cnt	Long	Peer quantity.

Parameter	Type	Description
status	String	Peer status. The value can be IsCreating , IsUpgrading , Adding/IsScaling , Isdeleting , Normal , AbNormal , or Unknown .
status_detail	String	Peer status in the format of 1/1. The denominator is the total number of peers in the organization, and the numerator is the number of normal peers.
pvc_name	String	Name of the PersistentVolumeClaim (PVC) used by the peer.
address	Array of PeerAddress objects	Peer domain name or IP address.

Table 5-37 CouchDBInfo

Parameter	Type	Description
user	String	CouchDB username.

Table 5-38 DmsKafkaInfo

Parameter	Type	Description
status	String	Status.
status_detail	String	Status details.
order_fade_enabled	Boolean	Whether to enable data aging on orderers.
addr	Array of strings	Kafka connection addresses.
order_fade_enabled	Boolean	Whether data aging on orderers is enabled if Kafka is used.
order_fade_cache	Long	Aging threshold (if data aging on orderers is enabled).

Table 5-39 lefInfo

Parameter	Type	Description
deploy_mode	Long	Organization-node relationship in edge cluster deployment mode. The value can be 0 (randomly matched) or 1 (bound).

Table 5-40 SfsInfo

Parameter	Type	Description
pvc_name	String	Name of the SFS file system.
name	String	Network storage name.
addr	String	Network storage address.
type	String	Network storage type.

Table 5-41 PeerAddress

Parameter	Type	Description
domain_port	String	Domain name address.
ip_port	String	IP address.

Table 5-42 OBSInfo

Parameter	Type	Description
name	String	OBS bucket name.
addr	String	OBS bucket address.

Status code: 400**Table 5-43** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-44 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-45 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-46 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-47 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "basic_info": {
    "id": "50ca216f-f0e5-ea05-5aee-de8d8f0fb6fa",
    "name": "bcs-vqf1x5",
    "version": "3.0.7",
    "service_type": "union",
    "purchase_type": "",
    "sign_algorithm": "ECDSA",
    "consensus": "sflic",
    "charging_mode": 1,
    "version_type": 3,
    "database_type": "goleveldb",
    "cluster_id": "aaace2ac-c18b-11ea-aefb-0255ac100022",
    "cluster_name": "roma-cluster",
    "cluster_type": "",
    "cluster_platform_type": "",
    "cluster_az": "",
    "created_time": "2020-07-22T04:03:44Z",
    "deploy_type": "",
    "order_fade_enabled": false,
    "order_fade_cache": 0,
    "is_cross_region": false,
    "is_support_rollback": false,
    "is_support_restful": false,
  }
}
```

```
"is_old_service" : false,
"old_service_version" : "",
"agent_portal_addr" : [ "24.68.213.148:30603" ],
"status" : "Normal",
"process_status" : "",
"order_status" : 0,
"deploy_status" : 0,
"block_info" : {
  "batch_timeout" : 2,
  "max_message_count" : 500,
  "preferred_maxbytes" : 2
},
"is_support_tc3" : false
},
"channels" : [ {
  "name" : "channel",
  "org_names" : [ "organization" ],
  "org_name_hash" : [ "organization:8ba7c7c2e84e57d1f16bd4739245e78aa2b08d3d" ],
  "peers" : {
    "8ba7c7c2e84e57d1f16bd4739245e78aa2b08d3d" :
[ "peer-8ba7c7c2e84e57d1f16bd4739245e78aa2b08d3d-0.peer-8ba7c7c2e84e57d1f16bd4739245e78aa2b08d3d.default.svc.cluster.local",
"peer-8ba7c7c2e84e57d1f16bd4739245e78aa2b08d3d-1.peer-8ba7c7c2e84e57d1f16bd4739245e78aa2b08d3d.default.svc.cluster.local" ]
  }
} ],
"peer_info" : [ {
  "name" : "organization",
  "node_cnt" : 2,
  "status" : "Normal",
  "status_detail" : "2/2",
  "pvc_name" : ""
} ],
"light_peer_info" : [ ],
"orderer_info" : {
  "name" : "bcs-vqf1x5-orderer",
  "node_cnt" : 4,
  "status" : "Normal",
  "status_detail" : "4/4",
  "address" : [ {
    "domain_port" :
"orderer-7d5c1e483233482cfe8c8b003e003da45f07d545-0.orderer-7d5c1e483233482cfe8c8b003e003da45f07d545.default.svc.cluster.local:7050",
    "ip_port" : "24.68.213.148:30835"
  }, {
    "domain_port" :
"orderer-7d5c1e483233482cfe8c8b003e003da45f07d545-1.orderer-7d5c1e483233482cfe8c8b003e003da45f07d545.default.svc.cluster.local:7050",
    "ip_port" : "24.68.213.148:30836"
  }, {
    "domain_port" :
"orderer-7d5c1e483233482cfe8c8b003e003da45f07d545-2.orderer-7d5c1e483233482cfe8c8b003e003da45f07d545.default.svc.cluster.local:7050",
    "ip_port" : "24.68.213.148:30837"
  }, {
    "domain_port" :
"orderer-7d5c1e483233482cfe8c8b003e003da45f07d545-3.orderer-7d5c1e483233482cfe8c8b003e003da45f07d545.default.svc.cluster.local:7050",
    "ip_port" : "24.68.213.148:30838"
  }
],
  "pvc_name" : ""
},
"couch_db_info" : {
  "user" : ""
},
"dms_kafka_info" : {
  "order_fade_enable" : false,
  "order_fade_cache" : 0,
  "status" : ""
}
```

```
{
  "status_detail" : ""
},
"ief_info" : {
  "deploy_mode" : 0
},
"sfs_info" : {
  "pvc_name" : "",
  "name" : "",
  "addr" : "",
  "type" : "host"
},
"agent_info" : {
  "name" : "baas-agent",
  "node_cnt" : 0,
  "status" : "Normal",
  "status_detail" : "1/1",
  "pvc_name" : ""
},
"restapi_info" : {
  "name" : "",
  "node_cnt" : 0,
  "status" : "",
  "status_detail" : "",
  "pvc_name" : ""
},
"tc3_taskserver_info" : {
  "name" : "",
  "node_cnt" : 0,
  "status" : "",
  "status_detail" : "",
  "pvc_name" : ""
},
"obs_bucket_info" : {
  "name" : "",
  "addr" : ""
}
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.4 Modifying a BCS Service

Function

This API is used to add peers, delete peers, add organizations, or delete organizations of a service. You can perform only one of these operations at a time. Adding and deleting peers are not supported for services deployed using IEF. This API is not applicable to services billed in yearly/monthly mode. The name of an IEF node can contain 4 to 24 characters.

URI

PUT /v2/{project_id}/blockchains/{blockchain_id}

Table 5-48 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-49 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-50 Request body parameters

Parameter	Mandatory	Type	Description
node_orgs	Yes	Array of NodeOrgs objects	Organizations to add peers for.
publicips	No	Array of IefNodeinfo objects	Information about the IEF node when you add an organization for a BCS service deployed using IEF. This parameter is required when the organization is bound with the node.
is_delete_org	No	Boolean	Indicates whether to delete organizations.

Table 5-51 NodeOrgs

Parameter	Mandatory	Type	Description
name	Yes	String	Organization name, which must be the same as the IEF node name if the organization is bound with a node. The name can contain 4 to 24 characters, including letters, digits, and hyphens (-). It cannot start with a hyphen (-).
node_count	Yes	Long	Number of peers in the organization. The value is an integer in the range from 1 to 2.
pvc_name	No	String	PVC name, which is required when you add an organization for a BCS service deployed using CCE.

Table 5-52 IefNodeinfo

Parameter	Mandatory	Type	Description
status	Yes	String	Node status, which must be ACTIVE if the node is to be used.
public_ip_address	Yes	String	Node EIP.
id	Yes	String	Node ID.

Response Parameters

Status code: 200

Table 5-53 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400

Table 5-54 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-55** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-56** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-57 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-58** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{
  "node_orgs" : [ {
    "name" : "organization",
    "node_count" : 1
  } ],
  "is_delete_org" : false
}
```

Example Responses

Status code: 200

Ok

```
{
  "operation_id" : "BCSSVC01-03-1617158790255323683"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006014",
  "error_msg" : "Invalid Parameter quotasType : ecs"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error CodesSee [Error Codes](#).

5.1.5 Creating Channels

Function

Creating Channels

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/channels

Table 5-59 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-60 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-61 Request body parameters

Parameter	Mandatory	Type	Description
channels	Yes	Array of ChannelCreateInfo objects	Channel list.

Table 5-62 ChannelCreateInfo

Parameter	Mandatory	Type	Description
channel_name	Yes	String	Channel name. The value can contain 4 to 24 characters, including lowercase letters and digits, and must start with a letter and end with a letter or digit.
channel_description	No	String	Channel description.

Response Parameters

Status code: 200

Table 5-63 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400

Table 5-64 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-65 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-66** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-67** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-68 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{
  "channels" : [ {
    "channel_name" : "channel_test",
    "channel_description" : ""
  } ]
}
```

Example Responses

Status code: 200

Ok

```
{
  "operation_id" : "BCSSVC01-03-1617158790255323683"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006014",
  "error_msg" : "Invalid Parameter quotasType : ecs"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.6 Querying Channel Information

Function

This API is used to query the channel details of a specified BCS service.

URI

GET /v2/{project_id}/blockchains/{blockchain_id}/channels

Table 5-69 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID

Request Parameters

Table 5-70 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200**Table 5-71** Response body parameters

Parameter	Type	Description
channels	Array of Channel objects	Channel information list.

Table 5-72 Channel

Parameter	Type	Description
name	String	Channel name.
description	String	Channel description.
create_time	String	Time when the channel was created.
consensus	String	Consensus mechanism.
peers	Map<String,Array<String>>	The key is the organization name and the value is the list of peer names.
consensusNodes	Map<String,Array<String>>	The key is the organization name and the value is the list of peer names.

Status code: 400

Table 5-73 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-74** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-75** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-76 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-77** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "channels" : [ {
    "name" : "channel",
    "peers" : {
      "organization" : [ "peer-328000cb35d8971e0b0388f6182f95eaa4100784-0",
"peer-328000cb35d8971e0b0388f6182f95eaa4100784-1" ]
    },
    "consensusNodes" : {
      "organization" : [ "peer-328000cb35d8971e0b0388f6182f95eaa4100784-0" ]
    },
    "consensus" : "etcdraft",
    "description" : "",
    "create_time" : "2020-10-15T10:24:51Z"
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401**Unauthorized**

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403**Forbidden**

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404**Not Found**

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500**InternalServerError**

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error CodesSee [Error Codes](#).

5.1.7 Adding Peers to a Channel

Function

This API is used to add peers to a channel. Currently, you can use this API to add peers to only one channel.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/channels/peers

Table 5-78 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-79 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-80 Request body parameters

Parameter	Mandatory	Type	Description
channel_peers	Yes	Array of PeerChannelInfo objects	Information about the peers that are added to a channel

Table 5-81 PeerChannelInfo

Parameter	Mandatory	Type	Description
channel_name	Yes	String	Name of the channel to which the peers are added.

Parameter	Mandatory	Type	Description
peers	Yes	Map<String,Integer>	Organization name and quantity of the peers being added. The key is the organization name and the value is the peer quantity.

Response Parameters

Status code: 200

Table 5-82 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400

Table 5-83 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-84 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-85** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-86** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-87** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "channel_peers" : [ {
```

```
"channel_name" : "channel02",  
  "peers" : {  
    "organization" : 2,  
    "organization2" : 1  
  }  
}]  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "operation_id" : "BCSSVC01-03-1617158790255323683"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006014",  
  "error_msg" : "Invalid Parameter quotasType : ecs"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.8 Removing Organizations from a Channel

Function

This API is used to remove organizations from a channel.

URI

PUT /v2/{project_id}/blockchains/{blockchain_id}/{channel_id}/orgs/quit

Table 5-88 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.
channel_id	Yes	String	Blockchain channel name.

Request Parameters

Table 5-89 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-90 Request body parameters

Parameter	Mandatory	Type	Description
org_names	Yes	Array of strings	Organization name list.

Response Parameters

Status code: 200

Table 5-91 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400

Table 5-92 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-93 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-94** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-95** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-96 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "org_names" : [ "organization", "organization2" ]  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "operation_id" : "BCSSVC01-06-1624275286218943067"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006014",  
  "error_msg" : "Invalid Parameter quotasType : ecs"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
}
```

```
{
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.9 Downloading Certificates

Function

This API is used to download the certificates of a specified BCS service.

URI

GET /v2/{project_id}/blockchains/{blockchain_id}/cert

Table 5-97 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID

Table 5-98 Query Parameters

Parameter	Mandatory	Type	Description
org_name	Yes	String	Name of the orderer or peer organization.
cert_type	Yes	String	Type of the certificate. Enumeration values: • admin • user • ca

Request Parameters

Table 5-99 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-100 Response body parameters

Parameter	Type	Description
-	File	Ok

Status code: 400

Table 5-101 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-102 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-103 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-104 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-105 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{  
  "result" : "success"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006005",  
  "error_msg" : "keyType error, keyType: blok not supported"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404**Not Found**

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500**InternalServerError**

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.10 Downloading the SDK Configuration

Function

This API is used to download the SDK configuration file of a specified BCS service.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/sdk-cfg

Table 5-106 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID

Request Parameters

Table 5-107 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Table 5-108 Request body parameters

Parameter	Mandatory	Type	Description
chaincode_name	Yes	String	Chaincode name, which can contain 6 to 25 lowercase letters or digits and must start with a letter.
cert_path	Yes	String	Directory for storing the SDK configuration file.
channel_name	Yes	String	Channel name.
peer_orgs	Yes	Map<String,Array<String>>	The key is the organization name and the value is the information of peers whose certificates are to be downloaded. Specify peers in ascending order, for example, 0, 1, and 2.
union_info	No	Map<String,Array<String>>	The key is the name of a consortium member and the value is the hash value array of the peer organization names of the member.

Parameter	Mandatory	Type	Description
is_multi_chan	No	Boolean	Indicates whether the request is a multi-channel request. If this parameter is set to true , configure channel_chaincode , and leave chaincode_name and channel_name empty.
channel_chain code	No	Map<String,Array<String>>	The key is the channel name and the value is the chaincode array corresponding to the channel.

Response Parameters

Status code: 200

Table 5-109 Response body parameters

Parameter	Type	Description
-	File	Ok

Status code: 400

Table 5-110 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-111 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-112** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-113** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-114 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{
  "chaincode_name" : "chaincode",
  "cert_path" : "/root",
  "channel_name" : "channel",
  "peer_orgs" : {
    "organization" :
    [ "peer-328000cb35d8971e0b0388f6182f95eaa4100784-0.peer-328000cb35d8971e0b0388f6182f95eaa4100784.default.svc.cluster.local",
      "peer-328000cb35d8971e0b0388f6182f95eaa4100784-1.peer-328000cb35d8971e0b0388f6182f95eaa4100784.default.svc.cluster.local" ]
  }
}
```

Example Responses

Status code: 200

Ok

```
{
  "result" : "success"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
}
```

```
{
  "error_msg": "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code": "BCS.4040404",
  "error_msg": "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code": "BCS.5000500",
  "error_msg": "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.11 Generating a User Certificate

Function

This API is used to generate a user certificate for a specific instance.

URI

```
POST /v2/{project_id}/blockchains/{blockchain_id}/orgs/{org_name}/usercert/
{user_name}
```

Table 5-115 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID
org_name	Yes	String	Peer organization name.
user_name	Yes	String	Username. The value can contain 4 to 24 characters, including lowercase letters and digits, and must start with a letter.

Request Parameters

Table 5-116 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Table 5-117 Request body parameters

Parameter	Mandatory	Type	Description
security_mode	No	Boolean	Indicates whether to enable the safe mode during certification generation. true : Enable the safe mode. Certificates are hosted by the system. Each username can be used to generate only one certificate. A maximum of 100 certificates can be generated by each organization. false : Do not enable the safe mode. Certificates are managed by users, and the quantity of certificates is unlimited.

Response Parameters

Status code: 200

Table 5-118 Response body parameters

Parameter	Type	Description
-	File	Ok

Status code: 400

Table 5-119 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-120 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-121 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-122** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-123** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "security_mode" : true  
}
```

Example Responses

Status code: 200

Ok

```
{
  "result" : "success"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized

Status Code	Description
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.12 Unfreezing a User Certificate

Function

This API is used to unfreeze the organization user certificate of a specified service instance. It takes 0.5 minutes to 1 minute for the unfreezing to take effect.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/orgs/{org_name}/usercert/{user_name}/unfreeze

Table 5-124 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
user_name	Yes	String	userName
blockchain_id	Yes	String	blockchainID
org_name	Yes	String	orgName

Request Parameters

Table 5-125 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Table 5-126 FormData parameters

Parameter	Mandatory	Type	Description
file	No	File	A .zip file of certificates. (Only one certificate file can be uploaded at a time, which is a .zip file less than 30 KB.)

Response Parameters

Status code: 200

Table 5-127 Response body parameters

Parameter	Type	Description
result	String	Operation result.

Status code: 400

Table 5-128 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-129 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-130** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-131** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-132** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Success

```
{
  "result" : "success"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Success
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.13 Freezing a User Certificate

Function

This API is used to freeze the organization user certificate of a specified service instance. It takes 0.5 minutes to 1 minute for the freezing to take effect.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/orgs/{org_name}/usercert/{user_name}/freeze

Table 5-133 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
user_name	Yes	String	userName
blockchain_id	Yes	String	blockchainID
org_name	Yes	String	orgName

Request Parameters

Table 5-134 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Table 5-135 FormData parameters

Parameter	Mandatory	Type	Description
file	No	File	A .zip file of certificates. (Only one certificate file can be uploaded at a time, which is a .zip file less than 30 KB.)

Response Parameters

Status code: 200

Table 5-136 Response body parameters

Parameter	Type	Description
result	String	Operation result.

Status code: 400

Table 5-137 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-138 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-139** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-140** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-141 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Success

```
{  
  "result" : "success"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006005",  
  "error_msg" : "keyType error, keyType: blok not supported"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Success
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error CodesSee [Error Codes](#).**5.1.14 Querying Quotas****Function**

This API is used to query the quotas of all BCS service resources under the specified project.

URI

GET /v2/{project_id}/quotas

Table 5-142 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-143 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-144 Response body parameters

Parameter	Type	Description
resources	Array of Resource objects	Quota information.

Table 5-145 Resource

Parameter	Type	Description
type	String	Resource type. The value can be instance (number of BCS services).
unit	String	Unit.
min	Long	Minimum limit of the quota.
max	Long	Maximum limit of the quota.
quota	Long	Quota.
used	Long	Used quota.
free	Long	Remaining quota.

Status code: 400

Table 5-146 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-147** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-148** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-149 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-150** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "resources" : [ {
    "type" : "instance",
    "unit" : "",
    "min" : 0,
    "max" : 100,
    "quota" : 5,
    "used" : 0,
    "free" : 0
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006014",
```

```
{
  "error_msg": "Invalid Parameter quotasType : ecs"
}
```

Status code: 401

Unauthorized

```
{
  "error_code": "BCS.4010401",
  "error_msg": "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code": "BCS.4030403",
  "error_msg": "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code": "BCS.4040404",
  "error_msg": "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code": "BCS.5000500",
  "error_msg": "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.15 Querying Flavors

Function

This API is used to query consortium blockchain flavors.

URI

GET /v2/{project_id}/blockchains/flavors

Table 5-151 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 5-152 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of returned records. Value range: 1–1000. Default value: 1000.
offset	No	Integer	Start position of a pagination query. The value must be a non-negative integer.

Request Parameters

Table 5-153 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-154 Response body parameters

Parameter	Type	Description
enterprise_spec	InstanceSpec object	Flavor of enterprise edition consortium blockchains.

Table 5-155 InstanceSpec

Parameter	Type	Description
org_peer_max_num	Long	Maximum number of peers allowed for an organization.
orderer_max_num	Long	Maximum number of orderers allowed for a consortium blockchain.
member_max_num	Integer	Maximum number of tenants allowed for a consortium blockchain.

Status code: 400**Table 5-156** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-157** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-158** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-159** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-160** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "enterprise_spec" : {
    "org_peer_max_num" : 5,
    "orderer_max_num" : 10,
    "member_max_num" : 50
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006005",
  "error_msg" : "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.16 Querying Peer Information

Function

This API is used to query the peer details of a specified BCS service.

URI

GET /v2/{project_id}/blockchains/{blockchain_id}/nodes

Table 5-161 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	blockchainID

Request Parameters

Table 5-162 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-163 Response body parameters

Parameter	Type	Description
node_orgs	Map<String, Org >	The key is the organization name and the value is the organization details.

Table 5-164 Org

Parameter	Type	Description
org_msp_id	String	Organization MSP ID.
org_domain	String	Organization domain name.
peers	Map<String, Node >	The key is the peer name and the value is the peer details.

Table 5-165 Node

Parameter	Type	Description
ip_port	String	Peer EIP information.
channels	Array of strings	Array of channels to which the peer is added.

Status code: 400

Table 5-166 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-167** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-168** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-169 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-170** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "node_orgs": {
    "organization": {
      "org_msp_id": "328000cb35d8971e0b0388f6182f95eaa4100784MSP",
      "org_domain": "peer-328000cb35d8971e0b0388f6182f95eaa4100784.default.svc.cluster.local",
      "peers": {
        "peer-328000cb35d8971e0b0388f6182f95eaa4100784-0": {
          "channels": [ "channel" ],
          "ip_port": "100.93.3.32:30605"
        },
        "peer-328000cb35d8971e0b0388f6182f95eaa4100784-1": {
          "channels": [ "channel" ],
          "ip_port": "100.93.3.32:30606"
        }
      }
    }
  }
}
```

```
}  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006005",  
  "error_msg" : "keyType error, keyType: blok not supported"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.17 Querying Asynchronous Operation Results

Function

Querying Asynchronous Operation Results

URI

GET /v2/{project_id}/operation/record

Table 5-171 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 5-172 Query Parameters

Parameter	Mandatory	Type	Description
blockchain_id	No	String	Blockchain ID.
operation_status	No	String	Operation status. Options (Information in the brackets describes the status.): waiting (to be processed), processing , finished (successful), failed , and stop (interrupted). Enumeration values: • waiting • processing • finished • failed • stop

Parameter	Mandatory	Type	Description
resource_type	No	String	Resource type. Options (Information in the brackets describes the type): BCSSVC01 (BCS), BCSSVC02 (UGBaaS), PLUGIN01 (RESTful API add-on), PLUGIN02 (trusted computing add-on), and PLUGIN03 (light-node add-on). Enumeration values: • BCSSVC01 • BCSSVC02 • PLUGIN01 • PLUGIN02 • PLUGIN03
operation_type	No	String	Operation type. Options (Information in the brackets describes the type.): 99 (OpCreate), 00 (OpDelete), 01 (OpUpgrade), 91 (OpUpgradeRb), 02 (OpAddOrg), 03 (OpScaleOrg), 04 (OpJoinChannel), and 05 (OpJoinUnion). Enumeration values: • 99 • 00 • 01 • 91 • 02 • 03 • 04 • 05
operation_id	No	String	Operation record ID.

Request Parameters

Table 5-173 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200**Table 5-174** Response body parameters

Parameter	Type	Description
count	Long	Total number of operation records.
operation_records	Array of RecordDetailInfo objects	Operation records.

Table 5-175 RecordDetailInfo

Parameter	Type	Description
operation_id	String	Operation record ID.
resource_type	String	Resource type.
operation_type	String	Operation type.
domain_id	String	Tenant ID.
project_id	String	Project ID.
blockchain_id	String	Blockchain ID.
blockchain_name	String	Blockchain name.
cluster_info	OprecordCluster object	Cluster information.
operation_process	Map<String, ProcessInfo >	Operation process. The key is the process name and the value is the process information.
record_time	Long	Time when the record was last modified.

Parameter	Type	Description
operation_status	String	Operation status.
message	Array of strings	Operation process information.
desc	String	Operation description.

Table 5-176 OprecordCluster

Parameter	Type	Description
cluster_type	String	Cluster type.
name	String	Cluster name.
cluster_id	String	Cluster ID.
node_infos	Array of NodeInfo objects	Node information.

Table 5-177 NodeInfo

Parameter	Type	Description
name	String	Node name.
id	String	Node ID.

Table 5-178 ProcessInfo

Parameter	Type	Description
step_name	String	Operation name.
status	String	Operation status.
detail	String	Operation details.
sub_steps	Map<String, SubDetail >	Sub-operation information in the format of "map[string][SubDetail]". The key is the sub-operation name and the value is the sub-operation result.
serial_num	Long	Serial number.

Table 5-179 SubDetail

Parameter	Type	Description
substep_name	String	Sub-operation name.
detail	String	Sub-operation details.
status	String	Sub-operation status.
message	Array of strings	Sub-operation process information.
start_time	Long	Start time.
end_time	Long	End time.
serial_num	Long	Serial number.

Status code: 400**Table 5-180** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-181** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-182 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-183** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-184** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "count" : 1,
  "operation_records" : [ {
    "operation_id" : "BCSSVC01-99-1618218591906093855",
    "resource_type" : "BCSSVC01",
    "operation_type" : "99",
    "domain_id" : "051ddda36880d3420f9bc01403fd8da0",
    "project_id" : "051ddda57400d2982fcc014fab5749d",
    "blockchain_id" : "23604f57-eac9-3777-e9f5-ae87457f2338",
    "blockchain_name" : "bcs-i3b0ms",
    "cluster_info" : {
      "cluster_type" : "cce",
      "name" : "asdfasf",
      "cluster_id" : "4df5652f-96dc-11eb-b49c-0255ac100b08",
      "node_infos" : null
    },
    "operation_process" : {
      "Create Other Dependent Resource" : {
        "step_name" : "Create Other Dependent Resource",
        "status" : "processing",
        "detail" : "",
        "sub_steps" : {
          "Create EIP" : {
            "substep_name" : "Create EIP",
            "detail" : null,
            "status" : "finished",
            "message" : null,
            "start_time" : 1618219038464462600,
            "end_time" : 1618219038464462600,
            "serial_num" : 1
          },
          "Create network storage" : {
            "substep_name" : "Create network storage",
            "detail" : "waiting",
            "status" : "",
            "message" : null,
            "start_time" : 0,
            "end_time" : 0,
            "serial_num" : 2
          }
        }
      },
      "serial_num" : 2
    },
    "Preparation" : {
      "step_name" : "Preparation",
      "status" : "finished",
      "detail" : "",
      "sub_steps" : {
        "Generate basic configurations" : {
          "substep_name" : "Generate basic configurations",
          "detail" : null,
          "status" : "finished",
          "message" : null,
          "start_time" : 1618218591951266000,
          "end_time" : 1618218591951266000,
          "serial_num" : 3
        }
      },
      "serial_num" : 1
    }
  },
  "record_time" : 1618218591906100000,
  "operation_status" : "processing",
}
```

```
"message" : null,  
"desc" : null  
}]  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006005",  
  "error_msg" : "keyType error, keyType: blok not supported"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found

Status Code	Description
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.18 Querying the BCS Service List

Function

This API is used to query the basic information of all BCS services under the specified project.

URI

GET /v2/{project_id}/blockchains

Table 5-185 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-186 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-187 Response body parameters

Parameter	Type	Description
blockchains	Array of BlockchainInfo objects	BCS service basic information.
count	Long	Total number of instances.

Table 5-188 BlockchainInfo

Parameter	Type	Description
id	String	BCS service instance ID.
name	String	Service name.

Status code: 400**Table 5-189** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-190** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-191** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-192** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-193** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "blockchains": [ {
    "id": "b268e1ca-b5c2-28f1-1e57-f571e14ecdbd",
    "name": "marbles"
  }, {
    "id": "75022dfb-e186-539e-cbc7-14020c934dd0",
    "name": "bcs-api"
  }, {
    "id": "04a33954-c707-4ab5-9d1a-b5e37624e56e",
    "name": "bcs-orgs"
  }, {
    "id": "4265b6ac-c7fd-b851-7dbc-1572364da341",
    "name": "bcs-dz8lo7"
  } ],
  "count": 3
}
```

Status code: 400

Bad Request

```
{
  "error_code": "BCS.4006005",
  "error_msg": "keyType error, keyType: blok not supported"
}
```

Status code: 401

Unauthorized

```
{
  "error_code": "BCS.4010401",
  "error_msg": "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code": "BCS.4030403",
  "error_msg": "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code": "BCS.4040404",
  "error_msg": "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code": "BCS.5000500",
  "error_msg": "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.19 Deleting a BCS Service

Function

This API is used to delete a BCS service. This API is not supported for BCS services billed on a yearly/monthly basis.

URI

DELETE /v2/{project_id}/blockchains/{blockchain_id}

Table 5-194 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Table 5-195 Query Parameters

Parameter	Mandatory	Type	Description
is_delete_storage	No	Boolean	
is_delete_obs	No	Boolean	

Parameter	Mandatory	Type	Description
is_delete_resource	No	Boolean	Whether to delete underlying CCE resources. This parameter is not required if the BCS service is deployed using IEF.
is_delete_ief	No	Boolean	Specifies whether to delete the underlying IEF edge nodes. This field is optional in CCE mode and mandatory in IEF mode.
is_delete_lightpeer	No	String	Optional in CCE mode. This parameter specifies whether to delete the IEF edge nodes used by the LightPeer add-on. To delete them, enter the corresponding edge node name. For example: is_delete_lightpeer=ief_node_name_1,ief_node_name_2.
ief_nodes_id	No	String	Optional in IEF mode. This parameter specifies whether to delete the IEF edge nodes used by edge clusters. To delete them, enter the edge node ID. For example: ief_nodes_id=1356f6f0-c448-4ec2-926f-c13b026369d9,0cdacbf9-2002-41e9-8123-2fa13e8d1449.

Request Parameters

Table 5-196 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-197 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400**Table 5-198** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-199** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-200** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-201 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-202 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "operation_id" : "BCSSVC01-03-1617158790255323683"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006014",
  "error_msg" : "Invalid Parameter is_delete_obs ,must be true or false"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.20 Removing a Peer from a Channel

Function

This API is used to remove a peer in an organization from a channel. If the target peer is the last one in the channel, call the **Removing Organization from a Channel** API to remove it.

URI

PUT /v2/{project_id}/blockchains/{blockchain_id}/{channel_id}/peers/quit

Table 5-203 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID. You can obtain it by calling the Querying the BCS Service List API.
channel_id	Yes	String	Blockchain channel name, which is the value of the name field in the channels returned by the Querying Service Information API.

Request Parameters

Table 5-204 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-205 Request body parameters

Parameter	Mandatory	Type	Description
org_name	Yes	String	Organization name. You can enter only one organization name.
peers	Yes	Integer	Number of peers to be removed. Value range: [0, total number of the organization peers].

Response Parameters

Status code: 200

Table 5-206 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400

Table 5-207 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-208 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-209** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-210** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-211** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "org_name" : "organization",  
  "peers" : 1  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "operation_id" : "BCSSVC01-03-1617158790255323683"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4001245",  
  "error_msg" : "The org name is invalid..."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.1.21 Deleting a Channel

Function

This API is used to delete a channel in a BCS service. Only empty channels can be deleted.

URI

DELETE /v2/{project_id}/blockchains/{blockchain_id}/channel/{channel_id}

Table 5-212 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID. You can obtain it by calling the Querying the BCS Service List API.
channel_id	Yes	String	Blockchain channel name, which is the value of the name field in the channels returned by the Querying Service Information API.

Request Parameters

Table 5-213 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Response Parameters

Status code: 200

Table 5-214 Response body parameters

Parameter	Type	Description
operation_id	String	Request result.

Status code: 400

Table 5-215 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-216 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-217** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-218** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-219** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "operation_id" : "BCSSVC01-03-1617158790255323683"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4001249",
  "error_msg" : "the channel(%s) is not empty for the bcs(%s), so cannot delete channel, please remove orgs from this channel first"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
}
```

```
"error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.2 BCS Consortium

5.2.1 Inviting Tenants to Join a Consortium

Function

This API is used to invite multiple tenants in batches to join a channel. This operation will send invitation notifications to the invitees.

URI

POST /v2/{project_id}/members/invitations

Table 5-220 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-221 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-222 Request body parameters

Parameter	Mandatory	Type	Description
bcs_id	Yes	String	BCS service ID of the inviting party.
channel_name	Yes	String	Name of the channel in the invitation.
invitor_username	No	String	Tenant name of the inviting party.
invited_userinfo	Yes	Array of InvitedDomain objects	List of invitees.

Table 5-223 InvitedDomain

Parameter	Mandatory	Type	Description
invited_user	Yes	String	Name of the invitee.

Response Parameters

Status code: 200

Table 5-224 Response body parameters

Parameter	Type	Description
result	String	Result of a successful request.

Status code: 400

Table 5-225 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-226** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-227** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-228 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-229** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{
  "bcs_id" : "ba0fc16d-c752-9ac4-7a43-7a84443e0a06",
  "channel_name" : "channel",
  "invitor_username" : "paas_bcs_c00218797_02",
  "invited_userinfo" : [ {
    "invited_user" : "xxx"
  } ]
}
```

Example Responses

Status code: 200

Ok

```
{
  "result" : "success"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006014",
}
```

```
{
  "error_msg": "Invalid Parameter quotasType : ecs"
}
```

Status code: 401

Unauthorized

```
{
  "error_code": "BCS.4010401",
  "error_msg": "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code": "BCS.4030403",
  "error_msg": "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code": "BCS.4040404",
  "error_msg": "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code": "BCS.5000500",
  "error_msg": "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.2.2 Processing an Invitation

Function

Processing an Invitation

URI

POST /v2/{project_id}/notification/handle

Table 5-230 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-231 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-232 Request body parameters

Parameter	Mandatory	Type	Description
channel_name	Yes	String	Channel in the invitation.
status	Yes	String	Invitation status. Enumeration values: - agreed - reject
invitor_info	Yes	HandleNotificationInvitor object	Inviting party information.
invitee_info	No	HandleNotificationInvitee object	Invitee information. This parameter is required when an invitation is accepted.

Parameter	Mandatory	Type	Description
invited_orgs	No	Array of HandleNotificationOrg objects	Organizations joining the consortium. This parameter is required when an invitation is accepted.

Table 5-233 HandleNotificationInvitor

Parameter	Mandatory	Type	Description
invitor_bcs_id	Yes	String	BCS service ID of the inviting party.
invitor_bcs_name	Yes	String	BCS service name of the inviting party.
invitor_project_id	Yes	String	Project ID of the inviting party.
invitor_user_id	Yes	String	Tenant ID of the inviting party. Log in to the BCS console as an inviting party, click the account name, and click My Credentials > API Credentials to obtain the Account ID .

Table 5-234 HandleNotificationInvitee

Parameter	Mandatory	Type	Description
invitee_bcs_id	Yes	String	BCS service ID of the invitee.
invitee_bcs_name	Yes	String	BCS service name of the invitee. This parameter is required when an invitation is accepted.
invitee_project_id	Yes	String	Project ID of the invitee.
invitee_user_id	Yes	String	Tenant ID of the invitee. Log in to the BCS console as an invitee, click the account name, and click My Credentials > API Credentials to obtain the Account ID .

Table 5-235 HandleNotificationOrg

Parameter	Mandatory	Type	Description
name	Yes	String	Organization that is used to join the consortium.

Response Parameters

Status code: 200

Table 5-236 Response body parameters

Parameter	Type	Description
result	String	Result of a successful request.

Status code: 400

Table 5-237 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-238 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-239** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-240** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-241** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "channel_name": "channel",
```

```
"invited_orgs" : [ {  
  "name" : "organization"  
} ],  
"invitor_info" : {  
  "invitor_user_id" : "051dded56100d4f90fe6c01d7d7a9240",  
  "invitor_bcs_id" : "6a9062bb-6aaf-a1cc-1756-7f45f9e0ea11",  
  "invitor_project_id" : "051dded75a80d4fb2f38c01d64a5789e",  
  "invitor_bcs_name" : "invitor-xxx"  
},  
"invitee_info" : {  
  "invitee_user_id" : "051ddda48580d3371fc6c014ba7175b5",  
  "invitee_bcs_id" : "6a9062bb-6aaf-a1cc-1756-7f45f9e0ea11",  
  "invitee_project_id" : "051dded75a80d4fb2f38c01d64a5789e",  
  "invitee_bcs_name" : "invitee-xxx"  
},  
"status" : "reject"  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "result" : "success"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006014",  
  "error_msg" : "Invalid Parameter quotasType : ecs"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.2.3 Querying All Notifications

Function

Querying All Notifications

URI

GET /v2/{project_id}/notifications

Table 5-242 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-243 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-244 Response body parameters

Parameter	Type	Description
notifications	Array of NotificationList objects	Notification list.

Table 5-245 NotificationList

Parameter	Type	Description
channel_name	String	Channel name.
status	String	Current status.
updated_time	String	Update time.
tc3_need	Boolean	Whether trusted computing is enabled.
invitor_info	InvitorInfo object	Inviting party information.
invitee_info	InviteeInfo object	Invitee information.
hide	Long	Whether the notification has been deleted.
invitee_orgs	Array of OrganizationV2 objects	Information about the invitee's organizations.
read_status	Long	Reading status.
cross_version_upgrade	String	Cross-version upgrade.

Table 5-246 InvitorInfo

Parameter	Type	Description
invitor_id	String	ID of the inviting party.
invitor_name	String	Username of the inviting party.
invitor_bcs_id	String	BCS service ID of the inviting party.
invitor_bcs_name	String	BCS service name of the inviting party.
invitor_consensus	String	Consensus mechanism used by the inviting party.
invitor_project_id	String	Project ID of the inviting party.
invitor_cluster_type	String	Cluster type used by the inviting party.
invitor_database_type	String	Database type used by the inviting party.
invitor_signature_algorithm	String	Signature algorithm used by the inviting party.
invitor_fabric_version	String	Fabric version used by the inviting party.
order_fade_enabled	Boolean	Whether to enable data aging on orderers.
order_fade_cache	Long	Aging threshold of orderer data.

Table 5-247 InviteeInfo

Parameter	Type	Description
invitee_id	String	ID of the invitee.
invitee_name	Long	Name of the invitee.
invitee_bcs_name	String	BCS service name of the invitee.
invitee_bcs_id	String	BCS service ID of the invitee.
invitee_project_id	String	Project ID of the invitee.

Table 5-248 OrganizationV2

Parameter	Type	Description
name	String	Organization name.
name_hash	String	Hash value of the organization name.
node_count	Integer	Peers in the organization.
status	String	Status.
status_detail	String	Status description.

Status code: 400**Table 5-249** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-250** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-251 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-252** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-253** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006014",
  "error_msg" : "Invalid Parameter quotasType : ecs"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.2.4 Listing Consortium Members

Function

Listing Consortium Members

URI

GET /v2/{project_id}/members

Table 5-254 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-255 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Response Parameters

Status code: 200

Table 5-256 Response body parameters

Parameter	Type	Description
members	Array of Member objects	Consortium member list.

Table 5-257 Member

Parameter	Type	Description
tcs_need	Boolean	Whether trusted computing is enabled.
channel_name	String	Channel name.
invited_orgs	Array of Organization V2 objects	Invitee's organizations.
invitor_info	MemberInvitor object	Inviting party information.
invitee_info	MemberInvitee object	Invitee information.

Table 5-258 OrganizationV2

Parameter	Type	Description
name	String	Organization name.
name_hash	String	Hash value of the organization name.
node_count	Integer	Peers in the organization.
status	String	Status.
status_detail	String	Status description.

Table 5-259 MemberInvitor

Parameter	Type	Description
invitor_bcs_id	String	BCS service ID of the inviting party.
invitor_bcs_name	String	BCS service name of the inviting party.
invitor_project_id	String	Project ID of the inviting party.
invitor_user_id	String	ID of the inviting party.
invitor_username	String	Tenant name of the inviting party.

Table 5-260 MemberInvitee

Parameter	Type	Description
invitee_bcs_id	String	BCS service ID of the invitee.
invitee_user_id	String	Tenant ID of the invitee.
invitee_username	String	Name of the invitee.

Status code: 400**Table 5-261** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-262** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-263 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-264** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-265** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

Ok

```
{
  "members": [ {
    "tcs_need": true,
    "channel_name": "channel",
    "invited_orgs": [ {
      "name": "organization",
      "name_hash": "25df6ee3d460a5daf6e11e6c5d945a43cda011fd",
      "node_count": 1,
      "status": "",
      "status_detail": ""
    } ],
    "invitor_info": {
      "invitor_bcs_id": "b268e1ca-b5c2-28f1-1e57-f571e14ecdbd",
      "invitor_bcs_name": "bcs-invitor",
      "invitor_project_id": "051ddda57400d2982fccc014fab5749d",
      "invitor_user_id": "28839632c7954c6380e59a4d416a60ee",
      "invitor_username": "test01"
    },
    "invitee_info": {
      "invitee_bcs_id": "05c0705f-2520-ff0f-3ae2-a48e7e3c8936",
      "invitee_user_id": "37556632c7954c6380e56c2e366a75se",
      "invitee_username": "test02"
    }
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code": "BCS.4006014",
  "error_msg": "Invalid Parameter quotasType : ecs"
}
```

Status code: 401

Unauthorized

```
{
  "error_code": "BCS.4010401",
  "error_msg": "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code": "BCS.4030403",
  "error_msg": "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code": "BCS.4040404",
  "error_msg": "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.2.5 Deleting Invitation Information

Function

This API is used to cancel invitations in batches or delete information about rejected or dismissed invitations, or invitations where the members have quit.

URI

DELETE /v2/{project_id}/members/invitations

Table 5-266 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-267 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-268 Request body parameters

Parameter	Mandatory	Type	Description
bcs_id	Yes	String	ID of the BCS service of the inviting party.
channel_name	Yes	String	Name of the channel in the invitation.
invited_userinfo	Yes	Array of InvitationDetail objects	List of invitees. The information can be queried using the ListMembers API. If an invitee has joined the consortium or the invitation status is released , enter the correct BCS service ID of the invitee and the invitation status.

Table 5-269 InvitationDetail

Parameter	Mandatory	Type	Description
invited_user	Yes	String	IAM user name of the invitee.
status	No	String	Invitation status. The options are as follows: quit , waiting , reject , and released . Invitation in other states cannot be deleted. Enumeration values: • quit • waiting • reject • released
invited_bcs_id	No	String	BCS service ID of the invitee.

Response Parameters

Status code: 200

Table 5-270 Response body parameters

Parameter	Type	Description
result	String	Result of a successful request.

Status code: 400

Table 5-271 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-272 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-273 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-274** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-275** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "bcs_id" : "83d85ea4-0f97-3bb7-07e2-4ad4b5a9f39d",  
  "channel_name" : "chan2",  
  "invited_userinfo" : [ {  
    "invited_user" : "bcstest02",
```

```
"status" : "quit",  
  "invited_bcs_id" : "11ba0ae0-2986-72d5-9049-43998c16fe86"  
}]  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "result" : "success"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4006014",  
  "error_msg" : "Invalid Parameter quotasType : ecs"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.2.6 Exiting a Consortium

Function

This API is used to exit a consortium.

URI

PUT /v2/{project_id}/members/quit

Table 5-276 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 5-277 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token. Minimum: 1 Maximum: 32768

Table 5-278 Request body parameters

Parameter	Mandatory	Type	Description
inviter_bcsid	Yes	String	BCS service ID of the inviting party. You can obtain it by calling the Querying the BCS Service List API.
inviter_project id	Yes	String	Project ID of the inviting party. Log in to the BCS console as an inviting party, click the account name, and click My Credentials > API Credentials to obtain the Project ID in the Projects area.
inviter_domai nid	Yes	String	Tenant ID of the inviting party. Log in to the BCS console as an inviting party, click the account name, and click My Credentials > API Credentials to obtain the Account ID .
inviter_userna me	Yes	String	Tenant name of the inviting party. Log in to the BCS console as an inviting party, click the account name, and click My Credentials > API Credentials to obtain the Account Name .
channel_nam e	Yes	String	Consortium channel name. On the BCS management plane, click Member Management to view the channel used in the invitation.
invitee_bcsid	Yes	String	BCS service ID of the invitee. You can obtain it by calling the Querying the BCS Service List API.
invitee_projec tid	Yes	String	Project ID of the invitee. Log in to the BCS console as an invitee, click the account name, and click My Credentials > API Credentials to obtain the Project ID in the Projects area.

Parameter	Mandatory	Type	Description
invitee_domainid	Yes	String	Tenant ID of the invitee. Log in to the BCS console as an invitee, click the account name, and click My Credentials > API Credentials to obtain the Account ID .

Response Parameters

Status code: 200

Table 5-279 Response body parameters

Parameter	Type	Description
result	String	Request result.
operation_id	String	Operation record ID.

Status code: 400

Table 5-280 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401

Table 5-281 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-282** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-283** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-284** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{
  "inviter_bcsid" : "a30962b6-xxxxx-f7ff1a0ec561",
  "inviter_projectid" : "b45a57xxxxx8a5",
  "inviter_domainid" : "a38c7xxxxd374d",
  "inviter_username" : "bcs-test",
  "channel_name" : "channel",
  "invitee_bcsid" : "e97ae066-xxxxx-f6bdb4fd3df1",
  "invitee_projectid" : "e64e7xxxxx65e1",
  "invitee_domainid" : "a38c7xxxxx374d"
}
```

Example Responses

Status code: 200

Ok

```
{
  "result" : "success",
  "operation_id" : "BCSSVC01-03-1617158790255323683"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4006003",
  "error_msg" : "Invalid Parameter notification status"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
}
```

```
{
  "error_msg": "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code": "BCS.5000500",
  "error_msg": "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.3 BCS Monitoring

5.3.1 Querying BCS Alarms

Function

This API is used to query the monitoring data of a specified metric of a BCS service.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/events

Table 5-285 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Parameter	Mandatory	Type	Description
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-286 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-287 Request body parameters

Parameter	Mandatory	Type	Description
metric_names	No	Array of strings	Metrics. Value range: cpuUsage : CPU usage** diskUsedRate **: disk usage memUsedRate : physical memory usage sendBytesRate : uplink rate (byte/s) recvBytesRate : downlink rate (byte/s) cpuCoreLimit : total CPU cores cpuCoreUsed : used CPU cores totalMem : total physical memory freeMem : available physical memory diskCapacity : disk capacity diskAvailableCapacity : available disk space Default value: the first five metrics.

Response Parameters

Status code: 200

Table 5-288 Response body parameters

Parameter	Type	Description
metrics	Array of MetricItemResultAPI objects	Request result.

Table 5-289 MetricItemResultAPI

Parameter	Type	Description
metric	MetricDemision object	Operation record ID.
dataPoints	Array of MetricDataPoints objects	Basic information of the monitoring data.

Table 5-290 MetricDemision

Parameter	Type	Description
namespace	String	Namespace.
metricName	String	Metric name.
dimensions	Array of Dimension objects	Dimension list.

Table 5-291 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Table 5-292 MetricDataPoints

Parameter	Type	Description
timestamp	Long	Timestamp.
unit	String	Metric unit.

Parameter	Type	Description
statistics	Array of StatisticValue objects	Statistical method.

Table 5-293 StatisticValue

Parameter	Type	Description
statistic	String	Statistical method.
value	Double	Statistics.

Status code: 400**Table 5-294** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-295** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403

Table 5-296 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-297** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-298** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "metric_names" : [ "cpuUsage", "diskUsedRate" ]  
}
```

Example Responses

Status code: 200

Ok

```
{
  "metrics" : [ {
    "metric" : {
      "namespace" : "abc",
      "metricName" : "def",
      "dimensions" : [ {
        "name" : "ghi",
        "value" : "lmn"
      } ]
    },
    "dataPoints" : [ {
      "timestamp" : "1467892800000",
      "unit" : "Percent",
      "statistics" : [ {
        "statistic" : "maximum",
        "value" : "23"
      } ]
    } ]
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4001108",
  "error_msg" : "metric names is wrong"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
}
```

```
"error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.3.2 Querying BCS Alarm Statistics

Function

This API is used to query the alarm statistics of a specified metric of a BCS service.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/events/statistic

Table 5-299 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-300 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-301 Request body parameters

Parameter	Mandatory	Type	Description
time_range	No	String	Value range of metrics.
metric_names	No	Array of strings	Metrics. Value range: cpuUsage : CPU usage** diskUsedRate **: disk usage memUsedRate : physical memory usage sendBytesRate : uplink rate (byte/s) recvBytesRate : downlink rate (byte/s) cpuCoreLimit : total CPU cores cpuCoreUsed : used CPU cores totalMem : total physical memory freeMem : available physical memory diskCapacity : disk capacity diskAvailableCapacity : available disk space Default value: the first five metrics.

Response Parameters

Status code: 200

Table 5-302 Response body parameters

Parameter	Type	Description
metrics	Array of MetricItemResultAPI objects	Request result.

Table 5-303 MetricItemResultAPI

Parameter	Type	Description
metric	MetricDemisi on object	Operation record ID.
dataPoints	Array of MetricDataP oints objects	Basic information of the monitoring data.

Table 5-304 MetricDemision

Parameter	Type	Description
namespace	String	Namespace.
metricName	String	Metric name.
dimensions	Array of Dimension objects	Dimension list.

Table 5-305 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Table 5-306 MetricDataPoints

Parameter	Type	Description
timestamp	Long	Timestamp.
unit	String	Metric unit.
statistics	Array of StatisticValu e objects	Statistical method.

Table 5-307 StatisticValue

Parameter	Type	Description
statistic	String	Statistical method.

Parameter	Type	Description
value	Double	Statistics.

Status code: 400**Table 5-308** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-309** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-310** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-311 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-312 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "metric_names" : [ "cpuUsage", "diskUsedRate" ]  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "metrics" : [ {
```

```
"metric" : {  
  "namespace" : "abc",  
  "metricName" : "def",  
  "dimensions" : [ {  
    "name" : "ghi",  
    "value" : "lmn"  
  } ]  
},  
"dataPoints" : [ {  
  "timestamp" : "1467892800000",  
  "unit" : "Percent",  
  "statistics" : [ {  
    "statistic" : "maximum",  
    "value" : "23"  
  } ]  
} ]  
} ]  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4001108",  
  "error_msg" : "metric names is wrong"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.3.3 Querying BCS Monitoring Data

Function

This API is used to query the monitoring data of a specified metric of a BCS service.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/metric/list

Table 5-313 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-314 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-315 Request body parameters

Parameter	Mandatory	Type	Description
metric_names	No	Array of strings	Metrics. Value range: cpuUsage : CPU usage** diskUsedRate **: disk usage memUsedRate : physical memory usage sendBytesRate : uplink rate (byte/s) recvBytesRate : downlink rate (byte/s) cpuCoreLimit : total CPU cores cpuCoreUsed : used CPU cores totalMem : total physical memory freeMem : available physical memory diskCapacity : disk capacity diskAvailableCapacity : available disk space Default value: the first five metrics.

Response Parameters

Status code: 200

Table 5-316 Response body parameters

Parameter	Type	Description
metrics	Array of MetricItemResultAPI objects	Request result.

Table 5-317 MetricItemResultAPI

Parameter	Type	Description
metric	MetricDemisi on object	Operation record ID.
dataPoints	Array of MetricDataP oints objects	Basic information of the monitoring data.

Table 5-318 MetricDemision

Parameter	Type	Description
namespace	String	Namespace.
metricName	String	Metric name.
dimensions	Array of Dimension objects	Dimension list.

Table 5-319 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Table 5-320 MetricDataPoints

Parameter	Type	Description
timestamp	Long	Timestamp.
unit	String	Metric unit.
statistics	Array of StatisticValu e objects	Statistical method.

Table 5-321 StatisticValue

Parameter	Type	Description
statistic	String	Statistical method.

Parameter	Type	Description
value	Double	Statistics.

Status code: 400**Table 5-322** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-323** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-324** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404

Table 5-325 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500

Table 5-326 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "metric_names" : [ "cpuUsage", "diskUsedRate" ]  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "metrics" : [ {
```

```
"metric" : {  
  "namespace" : "abc",  
  "metricName" : "def",  
  "dimensions" : [ {  
    "name" : "ghi",  
    "value" : "lmn"  
  } ]  
},  
"dataPoints" : [ {  
  "timestamp" : "1467892800000",  
  "unit" : "Percent",  
  "statistics" : [ {  
    "statistic" : "maximum",  
    "value" : "23"  
  } ]  
} ]  
} ]  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4001108",  
  "error_msg" : "metric names is wrong"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
  "error_msg" : "Internal Server Error"  
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.3.4 Listing Entity Monitoring Data of a BCS Service

Function

This API is used to query the monitoring data list of entities of a BCS service.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/entity/metric/list

Table 5-327 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-328 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-329 Request body parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Entity type. Options: org (peer organization) plugin (add-on) Default value: org .
entity_name	No	String	Entity name.

Response Parameters

Status code: 200

Table 5-330 Response body parameters

Parameter	Type	Description
metrics	Array of EntityMetricList objects	Request result.

Table 5-331 EntityMetricList

Parameter	Type	Description
dimensions	Array of Dimension objects	Request result.
values	Array of EntityMetricListItem objects	Monitoring data list.

Table 5-332 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Table 5-333 EntityMetricListItem

Parameter	Type	Description
cpuUsage	String	CPU usage.
diskReadRate	String	Disk read rate.
diskWriteRate	String	Disk write rate.
memUsage	String	Physical memory usage.
recvBytesRate	String	Downlink rate (bps).
sendBytesRate	String	Uplink rate (bps).
filesystemUsage	String	File system usage.

Status code: 400**Table 5-334** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-335** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-336** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-337** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-338** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "type" : "org",
```

```
{  
  "entity_name" : "baas-agent"  
}
```

Example Responses

Status code: 200

Ok

```
{  
  "metrics" : [ {  
    "dimensions" : [ {  
      "name" : "cluster",  
      "value" : "abc"  
    } ],  
    "values" : [ {  
      "cpuUsage" : 0.175,  
      "diskReadRate" : 0,  
      "diskWriteRate" : 0,  
      "memUsage" : 1.252,  
      "recvBytesRate" : 1703.617,  
      "sendBytesRate" : 2438.85,  
      "filesystemUsage" : 9.344  
    } ]  
  } ]  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "BCS.4004024",  
  "error_msg" : "invalid org name input"  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "BCS.4010401",  
  "error_msg" : "Incorrect token or token resolution failed"  
}
```

Status code: 403

Forbidden

```
{  
  "error_code" : "BCS.4030403",  
  "error_msg" : "No permissions to request this method"  
}
```

Status code: 404

Not Found

```
{  
  "error_code" : "BCS.4040404",  
  "error_msg" : "Not Found:the url is not found"  
}
```

Status code: 500

InternalServerError

```
{  
  "error_code" : "BCS.5000500",  
}
```

```
"error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

5.3.5 Querying Entity Monitoring Details of a BCS Service

Function

This API is used to query the monitoring details of entities of a specified BCS service.

URI

POST /v2/{project_id}/blockchains/{blockchain_id}/entity/instance/metric/list

Table 5-339 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
blockchain_id	Yes	String	BCS service ID.

Request Parameters

Table 5-340 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. You can obtain the token by calling the IAM API used to obtain a user token.

Table 5-341 Request body parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Entity type. [Options: org : baas-agent, orderer, and peer plugin : add-on] org is used by default.
entity_name	Yes	String	Names of the entities (baas-agent, orderer, and peer), which can be obtained in the Blockchain Organizations area on the service details page.
instance_name	Yes	String	Instance names. If the BCS service is deployed in a CCE cluster, the instance names are the workload (pod) name of each node. If the BCS service is deployed in an IEF cluster, go to the IEF console, choose Edge Applications > Containerized Applications to query the workload instance names.

Response Parameters

Status code: 200

Table 5-342 Response body parameters

Parameter	Type	Description
metrics	Array of MetricItemResultAPI objects	Request result.

Table 5-343 MetricItemResultAPI

Parameter	Type	Description
metric	MetricDemisi on object	Operation record ID.
dataPoints	Array of MetricDataP oints objects	Basic information of the monitoring data.

Table 5-344 MetricDemision

Parameter	Type	Description
namespace	String	Namespace.
metricName	String	Metric name.
dimensions	Array of Dimension objects	Dimension list.

Table 5-345 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Table 5-346 MetricDataPoints

Parameter	Type	Description
timestamp	Long	Timestamp.
unit	String	Metric unit.
statistics	Array of StatisticValu e objects	Statistical method.

Table 5-347 StatisticValue

Parameter	Type	Description
statistic	String	Statistical method.

Parameter	Type	Description
value	Double	Statistics.

Status code: 400**Table 5-348** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 401**Table 5-349** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 403**Table 5-350** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36

Parameter	Type	Description
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 404**Table 5-351** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Status code: 500**Table 5-352** Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
{  
  "type" : "org",  
  "entity_name" : "baas-agent",  
  "instance_name" : "baas-agent-868946786c-sswcn"  
}
```

Example Responses**Status code: 200**

Ok

```
{
  "metrics" : [ {
    "metric" : {
      "namespace" : "abc",
      "metricName" : "def",
      "dimensions" : [ {
        "name" : "ghi",
        "value" : "lmn"
      } ]
    },
    "dataPoints" : [ {
      "timestamp" : "1467892800000",
      "unit" : "Percent",
      "statistics" : [ {
        "statistic" : "maximum",
        "value" : "23"
      } ]
    } ]
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "BCS.4004021",
  "error_msg" : "GetOrgInstanceMetricDetail operation does not support IEF cluster type"
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "BCS.4010401",
  "error_msg" : "Incorrect token or token resolution failed"
}
```

Status code: 403

Forbidden

```
{
  "error_code" : "BCS.4030403",
  "error_msg" : "No permissions to request this method"
}
```

Status code: 404

Not Found

```
{
  "error_code" : "BCS.4040404",
  "error_msg" : "Not Found:the url is not found"
}
```

Status code: 500

InternalServerError

```
{
  "error_code" : "BCS.5000500",
  "error_msg" : "Internal Server Error"
}
```

Status Codes

Status Code	Description
200	Ok
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	InternalServerError

Error Codes

See [Error Codes](#).

6 Permissions Policies and Supported Actions

6.1 Introduction

This section describes fine-grained permissions management for your enhanced Hyperledger Fabric blockchains. If your Huawei Cloud account does not need individual IAM users, you may skip over this section.

A policy is a set of permissions defined in JSON format. By default, new IAM users do not have any permissions assigned. You need to add a user to one or more groups, and assign permissions policies to these groups. The user then inherits permissions from the groups it is a member of. This process is called authorization. After authorization, the user can perform specified operations on BCS 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.

For details about policies of enhanced Hyperledger Fabric blockchains, see [Permissions Management](#).

Supported Actions

Actions supported by policies are specific to APIs. Common concepts related to policies include:

- **Permissions:** Statements in a policy that allow or deny certain operations.
- **APIs:** APIs that will be called for performing certain operations.
- **Actions:** Specific operations that are allowed or denied.
- **Related actions:** Actions on which a specific action depends to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the related actions.

- Supported: IAM projects and enterprise projects

Type of projects in which policies can be used to grant permissions. A policy can be applied to IAM projects, enterprise projects, or both. 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 applied to IAM only. For details, see [What Are the Differences Between IAM and Enterprise Management?](#).

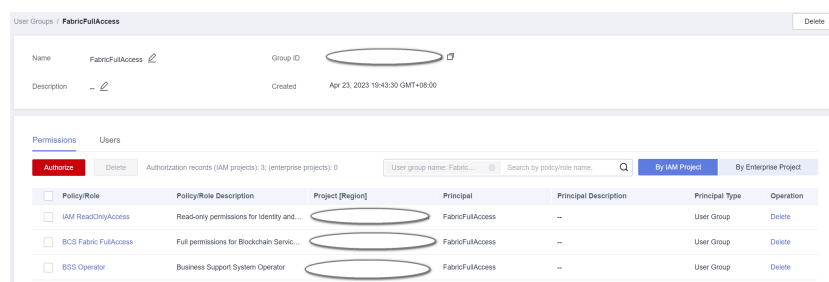
NOTE

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

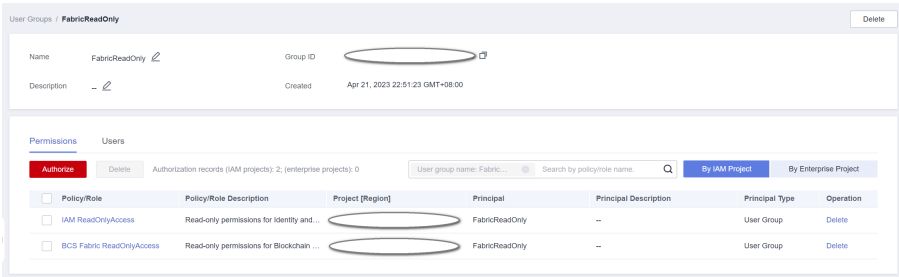
- Currently, only BCS instances created based on existing clusters support fine-grained authorization. To achieve this, grant namespace permissions to these clusters on the CCE console using an account with CCE Administrator or higher permissions. For example, grant the view namespace permission to the user group with the BCS Fabric ReadOnlyAccess permission and grant the cluster admin namespace permission to the user group with the BCS Fabric FullAccess permission. For details, see [Namespace Permissions \(Kubernetes RBAC-based\)](#).
- If a user wants to query the agency status, they must be granted the IAM ReadOnlyAccess permission. To authorize an agency, contact an account with the Security Administrator role.
- To create a yearly/monthly BCS instance, grant the BSS Operator role. Do not use the BCS Fabric ReadOnlyAccess and BSS roles together to avoid unexpected unsubscribing.
- Do not use the Tenant Guest role together with the BCS Fabric FullAccess policy, or some function pages will be hidden.
- Some APIs of dependent services do not support enterprise project-based permission management. You need to grant project permissions to IAM user groups during enterprise project authorization. For example, to create an instance, you need to grant the VPC ReadOnlyAccess project permission. To view monitoring information, you need to grant the AOM ReadOnlyAccess project permission. To view contract instances, you need to grant the CloudIDE ReadOnlyAccess project permission. To add an organization, you need to grant the SFS Turbo ReadOnlyAccess project permission.

Configure specific policies as required. The following figures show examples. For details, see [Creating a User Group and Assigning Permissions](#).

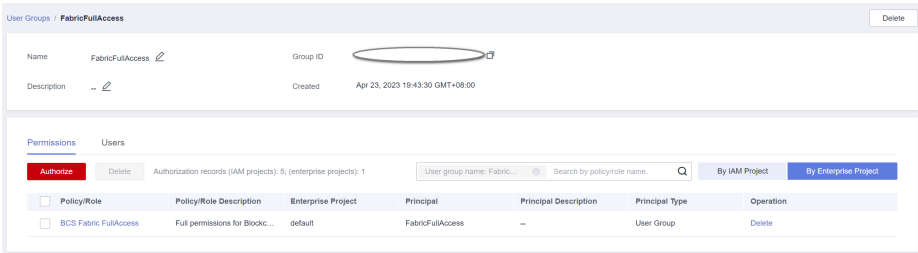
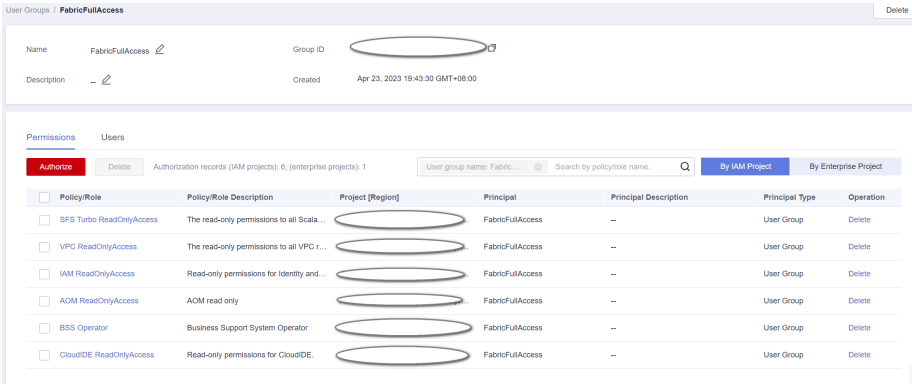
- After BCS Fabric FullAccess is granted to a project and the user group is configured.



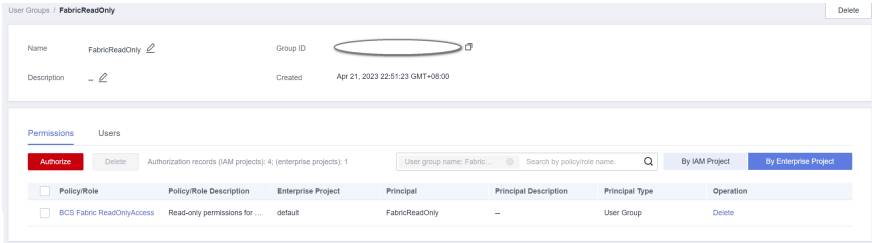
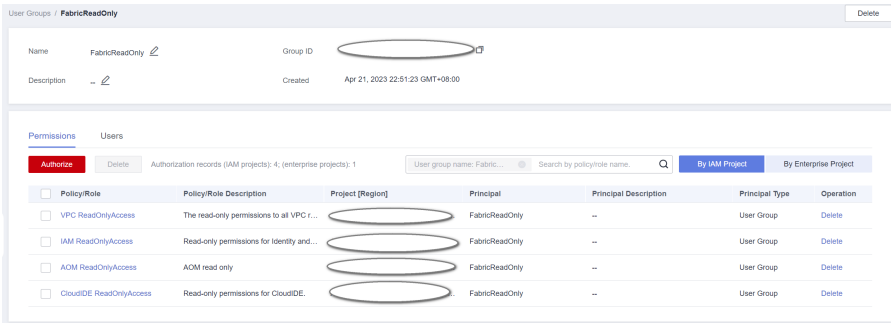
- After BCS Fabric ReadOnlyAccess is granted to a project and the user group is configured.



- After BCS Fabric FullAccess is granted to an enterprise project and the user group is configured.



- After BCS Fabric ReadOnlyAccess is granted to an enterprise project and the user group is configured.



6.2 Actions Supported by Custom Policies

The following introduces the actions supported by custom policies in enhanced Hyperledger Fabric blockchains.

NOTE

√ indicates that the item is supported, and × indicates that the item is not supported.

Table 6-1 Actions and system permissions of enhanced Hyperledger Fabric blockchains

Item	API	Action	IAM (Project)	Enterprise (Enterprise Project)
Querying Quotas	GET /v2/{project_id}/quotas	bcs:fabricInstance:listQuota	√	√
Querying Flavors	GET /v2/{project_id}/blockchains/flavors	bcs:fabricInstance:getFlavor	√	√
Querying Asynchronous Operation Results	GET /v2/{project_id}/operation/record	bcs:fabricInstance:listRecord	√	√
Creating a BCS Service	POST /v2/{project_id}/blockchains	bcs:fabricInstance:createOnDemand	√	√
Querying the BCS Service List	GET /v2/{project_id}/blockchains	bcs:fabricInstance:list	√	√
Querying Creation Status of a BCS Service	GET /v2/{project_id}/blockchains/{blockchain_id}/status	bcs:fabricInstance:getStatus	√	√
Querying Service Information	GET /v2/{project_id}/blockchains/{blockchain_id}	bcs:fabricInstance:getDetail	√	√

Item	API	Action	IAM (Project)	Enterprise (Enterprise Project)
Querying Peer Information	GET /v2/{project_id}/blockchains/{blockchain_id}/nodes	bcs:fabricInstance:getNodes	√	√
Modifying a BCS Service	PUT /v2/{project_id}/blockchains/{blockchain_id}	bcs:fabricInstance:update	√	√
Deleting Service Instances	DELETE /v2/{project_id}/blockchains/{blockchain_id}	bcs:fabricInstance:delete	√	√
Downloading Certificates	GET /v2/{project_id}/blockchains/{blockchain_id}/cert	bcs:fabricInstance:downloadCert	√	√
Downloading the SDK Configuration	POST /v2/{project_id}/blockchains/{blockchain_id}/sdk-cfg	bcs:fabricInstance:downloadSdkCfg	√	√
Generating a User Certificate	POST /v2/{project_id}/blockchains/{blockchain_id}/orgs/{org_name}/usercert/{user_name}	bcs:fabricInstance:createUserCert	√	√
Freezing a User Certificate	POST /v2/{project_id}/blockchains/{blockchain_id}/orgs/{org_name}/usercert/{user_name}/freeze	bcs:fabricInstance:freezeUserCert	√	√
Unfreezing a User Certificate	POST /v2/{project_id}/blockchains/{blockchain_id}/orgs/{org_name}/usercert/{user_name}/unfreeze	bcs:fabricInstance:unfreezeUserCert	√	√
Querying BCS Monitoring Data	POST /v2/{project_id}/blockchains/{blockchain_id}/metric/list	bcs:fabricInstance:listInstanceMetric	√	√

Item	API	Action	IAM (Project)	Enterprise (Enterprise Project)
Listing Entity Monitoring Data of a BCS Service	POST /v2/{project_id}/blockchains/{blockchain_id}/entity/metric/list	bcs:fabricInstance:listOrgMetric	√	√
Querying the Number of Monitored BCS Organization Instances	POST /v2/{project_id}/blockchains/{blockchain_id}/entity/instance/metric/list	bcs:fabricInstance:getOrgMetric	√	√
Creating Channels	POST /v2/{project_id}/blockchains/{blockchain_id}/channels	bcs:fabricChannel:create	√	√
Querying Channel Information	GET /v2/{project_id}/blockchains/{blockchain_id}/channels	bcs:fabricChannel:list	√	√
Adding Peers to a Channel	POST /v2/{project_id}/blockchains/{blockchain_id}/channels/peers	bcs:fabricChannel:addPeer	√	√
Removing a Peer from a Channel	PUT /v2/{project_id}/blockchains/{blockchain_id}/{channel_id}/peers/quit	bcs:fabricChannel:removePeer	√	√
Removing Organizations from a Channel	PUT /v2/{project_id}/blockchains/{blockchain_id}/{channel_id}/orgs/quit	bcs:fabricChannel:removeOrg	√	√
Deleting a Channel	DELETE /v2/{project_id}/blockchains/{blockchain_id}/channel/{channel_id}	bcs:fabricChannel:delete	√	√

Item	API	Action	IAM (Project)	Enterprise (Enterprise Project)
Inviting Tenants to Join a Consortium	POST /v2/{project_id}/members/invitations	bcs:fabricMember:createInvitation	√	√
Deleting Invitation Information	DELETE /v2/{project_id}/members/invitations	bcs:fabricMember:deleteInvitation	√	√
Listing Consortium Members	GET /v2/{project_id}/members	bcs:fabricMember:list	√	√
Exiting a Consortium	PUT /v2/{project_id}/members/quit	bcs:fabricMember:quit	√	√
Querying All Notifications	GET /v2/{project_id}/notifications	bcs:fabricNotification:list	√	√
Processing an Invitation	POST /v2/{project_id}/notification/handle	bcs:fabricNotification:handle	√	√

7 Appendix

7.1 Status Codes

Table 7-1 describes status codes.

Table 7-1 Status codes

Status Code	Message	Description
100	Continue	The server has received the initial part of the request and the client should continue to send the remaining part. It is issued on a provisional basis while request processing continues. It alerts the client to wait for a final response.
101	Switching Protocols	The requester has asked the server to switch protocols and the server has agreed to do so. The protocol can only be switched to a more advanced protocol. For example, the current HTTPS protocol is switched to a later version.
200	Success	The request is successfully sent.
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 server successfully processed the request, but is returning information that may be from another source.

Status Code	Message	Description
204	NoContent	The request has been fulfilled, but the HTTPS response does not contain a response body. The status code is returned in response to an HTTPS OPTIONS request.
205	Reset Content	The server has successfully processed the request, but does not return any content. Unlike a 204 response, this response requires that the requester 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 should be permanently directed 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 a different URI, and should be retrieved using a GET or POST method.
304	Not Modified	The requested resource has not been modified. In such case, there is no need to retransmit the resource since the client still has a previously-downloaded copy.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	The HTTPS status code is no longer used.
400	BadRequest	The request is invalid. 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	This status code is reserved for future use.

Status Code	Message	Description
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.
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 waiting for the request. The client may re-initiate the request without modifications at any later time.
409	Conflict	The request cannot be processed due to a conflict. This status code indicates that the resource that the client attempts to create already exists, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource is no longer available. The requested resource has been deleted permanently.
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 disable the connection to prevent the client from sending requests consecutively. If the server temporarily cannot process the request, the response will contain a Retry-After header field.

Status Code	Message	Description
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.
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 repeat 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 acting as a gateway or proxy receives an invalid response from a remote server.
503	ServiceUnavailable	The requested service is unavailable. The client should modify the request instead of re-initiating it.
504	ServerTimeout	The server could not return a timely response. The response will reach the client only if the request carries the timeout parameter.
505	HTTPS Version not supported	The server does not support the HTTPS protocol version used in the request.

7.2 Error Codes

If an error code starting with APIGW is returned after you call an API, rectify the fault by referring to the instructions provided in [API Gateway Error Codes](#).

Status Code	Error Codes	Error Message	Description	Solution
100	BCS.4006012	Invalid channel name.	Invalid channel name.	Enter a valid channel name.
400	BCS.2001001	BCS service modified.	BCS service modified.	The BCS service can work properly.
400	BCS.4001001	Operation failed. The BCS service is being created.	Operation failed. The BCS service is being created.	Wait until the service is created.
400	BCS.4001002	Operation failed. The BCS service is being upgraded.	Operation failed. The BCS service is being upgraded.	Wait until the upgrade is complete.
400	BCS.4001003	Operation failed. The BCS service is being deleted.	Operation failed. The BCS service is being deleted.	Wait until the deletion is complete.
400	BCS.4001006	Failed to delete the SFS file system because the CCE API or the cluster status is abnormal.	Failed to delete the SFS file system because the CCE API or the cluster status is abnormal.	Manually delete the SFS file system. Procedure: Log in to the CCE console, choose Resource Management > Storage , and locate the file system in the cluster corresponding to the BCS service. For an SFS file system, click Delete in the Operation column. For an SFS Turbo file system, click Unbind and then go to the SFS console to delete the file system.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001007	Services billed in the yearly/ monthly mode cannot be deleted.	Services billed in the yearly/ monthly mode cannot be deleted.	Services billed in the yearly/ monthly mode cannot be deleted. They can only be unsubscribed from on the console.
400	BCS.4001008	Operation failed. Bind an EIP to at least one node in the cluster and try again.	Operation failed. Bind an EIP to at least one node in the cluster and try again.	Bind an EIP to at least one node in the cluster and try again. Procedure: Log in to the CCE console. Choose Resource Management > Nodes . Locate the nodes in the cluster corresponding to the service, and perform the following steps on any node: On the node details page, click the node name. On the displayed ECS console, click the ECS name. On the ECS details page, click the EIPs tab. Click Bind EIP .

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001009	Operation failed. Bind an EIP to each node in the cluster and try again.	Operation failed. Bind an EIP to each node in the cluster and try again.	Bind an EIP to each node in the cluster and try again. Procedure: Log in to the CCE console. Choose Resource Management > Nodes . In the cluster corresponding to the service, locate the nodes that have not been bound with EIPs, and perform the following steps on each node: On the node details page, click the node name. On the displayed ECS console, click the ECS name. On the ECS details page, click the EIPs tab. Click Bind EIP .
400	BCS.4001010	Failed to download the BCS certificate because the etcd connection is abnormal.	Failed to download the BCS certificate because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001011	Operation failed. The service does not exist or has been deleted. Refresh the page.	Operation failed. The service does not exist or has been deleted. Refresh the page.	Wait for a few moments and refresh the page.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001015	Failed to obtain the peer information because the etcd connection is abnormal.	Failed to obtain the peer information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001018	Operation failed. A service with the same name already exists. Enter another name and try again.	Operation failed. A service with the same name already exists. Enter another name and try again.	Change the service name and try again.
400	BCS.4001019	Failed to obtain the image version for upgrade because the required image is not in the image repository or the version configuration is incorrect.	Failed to obtain the image version for upgrade because the required image is not in the image repository or the version configuration is incorrect.	Check the service version configuration and ensure that the image required for the upgrade is in the image repository.
400	BCS.4001020	Operation failed. The service status is abnormal or the service is being processed. Try again later.	Operation failed. The service status is abnormal or the service is being processed. Try again later.	Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001021	Failed to add the peer to the channel because the CCE API, cluster status, or etcd connection is abnormal.	Failed to add the peer to the channel because the CCE API, cluster status, or etcd connection is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001025	Invalid EIP. The EIP may have been unbound from the cluster. Check the EIP binding of the cluster.	Invalid EIP. The EIP may have been unbound from the cluster. Check the EIP binding of the cluster.	Log in to the CCE console, choose Resource Management > Clusters , and click the cluster corresponding to the service. Check whether the cluster nodes have been bound with EIPs. If not, bind EIPs to the cluster nodes.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001026	Failed to save the EIP to etcd because the etcd connection is abnormal.	Failed to save the EIP to etcd because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001027	Failed to update the EIP information in the consortium member configuration because the CCE API, cluster status, or etcd connection is abnormal.	Failed to update the EIP information in the consortium member configuration because the CCE API, cluster status, or etcd connection is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001030	Operation failed because the etcd connection check failed.	Operation failed because the etcd connection check failed.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001031	Failed to query the member details because the etcd connection is abnormal.	Failed to query the member details because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001032	You do not have the required permission.	You do not have the required permission.	Configure the permissions by following the instructions provided in <i>BCS User Guide > Service Deployment > Prerequisites</i> .
400	BCS.4001033	Failed to delete the member information because the etcd connection is abnormal.	Failed to delete the member information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001036	Operation failed. The notification does not exist. Refresh the page.	Operation failed. The notification does not exist. Refresh the page.	Refresh the page later.
400	BCS.4001040	Failed to decline the invitation for joining a channel. Refresh the page and check the invitation status later.	Failed to decline the invitation for joining a channel. Refresh the page and check the invitation status later.	Refresh the page and check the invitation status later.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001042	Failed to query the member information because the etcd connection is abnormal.	Failed to query the member information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001043	Failed to modify the member information because the etcd connection is abnormal.	Failed to modify the member information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001044	Failed to accept the invitation. The acceptance is being processed. Refresh the page and check the invitation status later.	Failed to accept the invitation. The acceptance is being processed. Refresh the page and check the invitation status later.	Refresh the page later to view the processing result.
400	BCS.4001045	Weak password.	Weak password.	Enter a strong password.
400	BCS.4001047	Failed to modify the etcd data because the etcd connection is abnormal.	Failed to modify the etcd data because the etcd connection is abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001048	Failed to change the password because the CCE cluster is abnormal.	Failed to change the password because the CCE cluster is abnormal.	Log in to the CCE console and choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster nodes are normal. If they are abnormal, rectify the fault by referring to the CCE troubleshooting guide. After the cluster restores, try again. If the cluster nodes have been deleted, buy nodes and create a BCS service again.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001049	Operation failed. The cluster status is abnormal. Check the cluster status and try again.	Operation failed. The cluster status is abnormal. Check the cluster status and try again.	Log in to the CCE console and choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster nodes are normal. If they are abnormal, rectify the fault by referring to the CCE troubleshooting guide. After the cluster restores, try again. If the cluster nodes have been deleted, buy nodes and create a BCS service again.
400	BCS.4001050	Scaling failed because the CCE API or the cluster status is abnormal.	Scaling failed because the CCE API or the cluster status is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001051	Scaling failed due to incorrect request parameters. Check whether the number of peers in the organization has reached the limit (5).	Scaling failed due to incorrect request parameters. Check whether the number of peers in the organization has reached the limit (5).	Ensure that the number of peers in each organization does not exceed the upper limit (5) and try again.
400	BCS.4001052	Operation failed because it timed out.	Operation failed because it timed out.	View the result later.
400	BCS.4001053	Failed to query the network storage because the CCE API or the cluster status is abnormal.	Failed to query the network storage because the CCE API or the cluster status is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001054	Failed to add the peer to the channel because the CCE cluster status or etcd connection is abnormal.	Failed to add the peer to the channel because the CCE cluster status or etcd connection is abnormal.	Log in to the CCE console and check whether CCE is normal. Choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001058	Failed to delete the member information because the etcd connection may be abnormal.	Failed to delete the member information because the etcd connection may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001062	The maximum number of tenants that can be invited has been reached.	The maximum number of tenants that can be invited has been reached.	Do not invite more tenants after the invitation limit has been reached.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001063	The maximum number of invitation notifications that can be received has been reached.	The maximum number of invitation notifications that can be received has been reached.	Log in to the BCS console, go to the Notification Management page, and check whether the number of received invitations reaches the upper limit (100). Process the received notifications.
400	BCS.4001067	Failed to delete the notification. The etcd connection may be abnormal.	Failed to delete the notification. The etcd connection may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001069	The chaincode is being instantiated. Try again later.	The chaincode is being instantiated. Try again later.	Log in to the Chaincode Management page of the corresponding service to check the chaincode instantiation status. After the instantiation is complete, perform the operation again.
400	BCS.4001078	Failed to query the channel information because the etcd connection may be abnormal.	Failed to query the channel information because the etcd connection may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001080	Operation failed because the CCE cluster information failed to be obtained.	Operation failed because the CCE cluster information failed to be obtained.	Check if CCE is normal by trying to log in to the CCE console. If the login is successful, choose Resource Management > Clusters to check whether the cluster used by the service is normal. If the cluster is abnormal, try again after it restores.
400	BCS.4001081	Failed to obtain the invitee's organization because the etcd is abnormal.	Failed to obtain the invitee's organization because the etcd is abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001086	Permission verification failed.	Permission verification failed.	Configure the permissions by following the instructions provided in <i>BCS User Guide > Service Deployment > Prerequisites</i> .
400	BCS.4001088	Operation failed due to insufficient quota. Increase the quota and try again.	Operation failed due to insufficient quota. Increase the quota and try again.	Contact the administrator to increase the quota on ManageOne.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001089	Deletion failed. Failed to delete the BCS information.	Deletion failed. Failed to delete the BCS information.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001090	Operation failed. No node is available in the cluster. Try again later.	Operation failed. No node is available in the cluster. Try again later.	Check whether the node is abnormal or has been deleted. Log in to the CCE console. Choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster has nodes. If not, the nodes may have been deleted. In this case, buy nodes and create a BCS service again. If the nodes are abnormal, restore the nodes as instructed by the CCE troubleshooting document and try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001095	Failed to delete block configurations due to etcd exceptions.	Failed to delete block configurations due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001096	Failed to delete the AgentConfig information due to etcd exceptions.	Failed to delete the AgentConfig information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001097	Failed to delete the CCE cluster occupation information due to etcd exceptions.	Failed to delete the CCE cluster occupation information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001098	Failed to delete the SDK information due to etcd exceptions.	Failed to delete the SDK information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001099	Failed to delete the channel information due to etcd exceptions.	Failed to delete the channel information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001100	Failed to delete the EIP information due to etcd exceptions.	Failed to delete the EIP information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001101	Failed to delete the port information due to etcd exceptions.	Failed to delete the port information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001102	Failed to delete the ipmapping information due to etcd exceptions.	Failed to delete the ipmapping information due to etcd exceptions.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001104	Failed to delete the deployment template because the CCE cluster status may be abnormal.	Failed to delete the deployment template because the CCE cluster status may be abnormal.	Check the CCE cluster status. Log in to the CCE console and choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, restore the cluster as instructed by the <i>CCE User Guide</i> and then try again.
400	BCS.4001105	Deleting the deployment template timed out because the CCE cluster status may be abnormal.	Deleting the deployment template timed out because the CCE cluster status may be abnormal.	Check the CCE cluster status. Log in to the CCE console and choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, restore the cluster as instructed by the <i>CCE User Guide</i> and then try again.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001106	Operation failed. The cluster does not exist. Try again later.	Operation failed. The cluster does not exist. Try again later.	Log in to the CCE console and choose Resource Management > Clusters . Check if the cluster used by the BCS service exists. If the cluster does not exist, create a cluster and buy a BCS service again.
400	BCS.4001109	Operation failed. Failed to obtain the peer organization status. Try again later.	Operation failed. Failed to obtain the peer organization status. Try again later.	Try again later.
400	BCS.4001110	Failed to create CertBitConf because the network may be faulty.	Failed to create CertBitConf because the network may be faulty.	Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001120	Failed to delete the service. The member failed to exit the consortium. The cluster or etcd connection is abnormal.	Failed to delete the service. The member failed to exit the consortium. The cluster or etcd connection is abnormal.	Log in to the CCE console and check whether CCE is normal. Choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001123	Failed to delete the service because the cluster has been hibernated.	Failed to delete the service because the cluster has been hibernated.	Wake up the cluster and try again. Log in to the CCE console and choose Resource Management > Clusters . Locate the cluster used by the service, choose More > Wake . After the cluster status is restored, delete the BCS service again.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001130	Failed to save the node and port information because the etcd, CCE API, or cluster is abnormal.	Failed to save the node and port information because the etcd, CCE API, or cluster is abnormal.	Log in to the CCE console and check whether CCE is normal. Choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001133	Failed to save the channel information because etcd may be abnormal.	Failed to save the channel information because etcd may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001134	Failed to save the BCS information because etcd may be abnormal.	Failed to save the BCS information because etcd may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001135	Failed to save the certificate because etcd may be abnormal.	Failed to save the certificate because etcd may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001136	Failed to save bitconf because etcd may be abnormal.	Failed to save bitconf because etcd may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001137	Failed to save the deployment template because etcd may be abnormal.	Failed to save the deployment template because etcd may be abnormal.	Check the etcd status. If etcd is abnormal, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001149	Failed to query the cluster details because the CCE service or API is abnormal.	Failed to query the cluster details because the CCE service or API is abnormal.	Check the CCE service and API status. Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001161	Failed to add the organization because the etcd connection or CCE API may be abnormal.	Failed to add the organization because the etcd connection or CCE API may be abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001171	Weak password.	Weak password.	Enter a strong password.
400	BCS.4001196	Operation failed because the current service or services in the consortium are being updated. Try again later.	Operation failed because the current service or services in the consortium are being updated. Try again later.	Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001202	Deletion failed. The service does not exist or has already been deleted. Refresh the page.	Deletion failed. The service does not exist or has already been deleted. Refresh the page.	Refresh the Service Management page.
400	BCS.4001203	Operation failed. Failed to obtain the deployment task.	Operation failed. Failed to obtain the deployment task.	Try again.
400	BCS.4001212	Operation failed. The ECS specified by node_flavor in the request cannot be purchased.	Operation failed. The ECS specified by node_flavor in the request cannot be purchased.	Specify an ECS with other flavors in the node_flavor.
400	BCS.4001234	Operation failed because the BCS service is in the Hibernating or Frozen state.	Operation failed because the BCS service is in the Hibernating or Frozen state.	Click More > Wake in the Operation column of the row containing the specified service and try again after the service is normal.
400	BCS.4001242	Failed to remove the organization from a channel.	Failed to remove the organization from a channel.	Remove the organization again until the channel and service statuses are normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001301	Failed to query the ROMA instance information because the ROMA instance may have been deleted or its status is abnormal.	Failed to query the ROMA instance information because the ROMA instance may have been deleted or its status is abnormal.	Log in to the ROMA console and check whether the instance status is normal. Wait until the instance status is normal or use an unused, normal instance.
400	BCS.4001302	Failed to query the cluster information because the cluster has been deleted or is abnormal.	Failed to query the cluster information because the cluster has been deleted or is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . If the cluster corresponding to the BCS service does not exist, the cluster has been deleted. In this case, use an existing, normal cluster to buy a BCS service again. If the cluster exists, check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored or use a normal cluster to create another BCS service.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001304	Failed to query the ROMA instance list because the ROMA MQS API is abnormal.	Failed to query the ROMA instance list because the ROMA MQS API is abnormal.	Log in to the ROMA console and check whether ROMA MQS and the ROMA instance is normal. If they are abnormal, try again after they are normal or rectify the fault as instructed by the ROMA troubleshooting document.
400	BCS.4001305	Failed to obtain the topic information of the ROMA instance because the ROMA instance is abnormal or has been deleted.	Failed to obtain the topic information of the ROMA instance because the ROMA instance is abnormal or has been deleted.	Log in to the ROMA console to check whether the ROMA instance used by the BCS service is normal. If it is abnormal, try again after it is normal. If the instance is deleted, purchase a new ROMA instance and create a BCS service again.
400	BCS.4001306	Operation failed because the ROMA instance is currently in use.	Operation failed because the ROMA instance is currently in use.	Select an unused, normal ROMA instance.
400	BCS.4001307	Operation failed because you do not have the permissions to view DMS data. Check your permissions in IAM.	Operation failed because you do not have the permissions to view DMS data. Check your permissions in IAM.	Check the account permissions in IAM. For details about the required permissions, see the user guide. Add the DMS permissions and try again.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001356	Operation failed. No network storage space has been configured for the cluster.	Operation failed. No network storage space has been configured for the cluster.	Buy a BCS service again. When configuring Network Storage , choose Create SFS File System .

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001401	Failed to update the IPMapping information of the current tenant because etcd, the CCE API, or the cluster is abnormal.	Failed to update the IPMapping information of the current tenant because etcd, the CCE API, or the cluster is abnormal.	Log in to the CCE console and choose Resource Management > Clusters . Locate the cluster used by the BCS service, and check whether the cluster is normal. If the cluster is abnormal, rectify the fault by referring to the CCE troubleshooting document or use an available cluster to buy another BCS service. If the cluster status is normal, choose Configuration Center > ConfigMaps . Filter the configuration items by cluster, find the IPMapping of the cluster corresponding to the BCS service, and check whether the configuration is correct and whether the format is complete. If no fault is found in the preceding steps, rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try

Status Code	Error Codes	Error Message	Description	Solution
				again after etcd is normal.
400	BCS.4001402	Failed to update the access address because the EIP information in the SDK configuration fails to be updated due to an etcd exception.	Failed to update the access address because the EIP information in the SDK configuration fails to be updated due to an etcd exception.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001404	Failed to update the EIP information of other consortium members because etcd may be abnormal.	Failed to update the EIP information of other consortium members because etcd may be abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001407	Try again or check whether ETCD is normal.	Obtain EIP data failed.	Try again or check whether ETCD is normal.
400	BCS.4001408	Do not use the domain name repeatedly. Apply for a new domain name or add a new domain name resolution set.	The domain name has been used in other BCS instances.	Do not use the domain name repeatedly. Apply for a new domain name or add a new domain name resolution set.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001520	Failed to obtain the topology information of the invitee because etcd may be abnormal.	Failed to obtain the topology information of the invitee because etcd may be abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001521	Failed to obtain the topology information of the inviting party because etcd may be abnormal.	Failed to obtain the topology information of the inviting party because etcd may be abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001708	Operation failed. The IP address of the IEF node cluster is incorrect.	Operation failed. The IP address of the IEF node cluster is incorrect.	1. Log in to the CCE console. 2. Choose Resource Management > Clusters, and click the cluster of a target BCS service. 3. Click Nodes, view the EIP bound to the cluster node. 4. Set the public IP addresses of these IEF nodes to the EIPs bound to the nodes.
400	BCS.4001717	Operation failed. The number of IEF nodes is less than that required by the consensus policy.	Operation failed. The number of IEF nodes is less than that required by the consensus policy.	Buy IEF nodes required by the consensus policy or buy more such nodes.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001750	Failed to query the quota from ManageOne because the ManageOne API may be abnormal.	Failed to query the quota from ManageOne because the ManageOne API may be abnormal.	Check the ManageOne API status. Log in to ManageOne and go to the quota management page. If the displayed page shows the service quotas in the specified region, the ManageOne API is normal. If the ManageOne API is abnormal, try again after it restores.
400	BCS.4001751	The BCS service quota is not found because the quota is not registered or the quota configuration is incorrect.	The BCS service quota is not found because the quota is not registered or the quota configuration is incorrect.	Log in to ManageOne and go to the quota management page. If the displayed page shows the BCS quota in the specified region, the BCS quota has been registered correctly. Otherwise, the BCS quota has not been registered or the quota configuration is incorrect. In this case, contact the administrator to register the BCS quota again.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001752	Failed to modify the quota on ManageOne because the ManageOne API may be abnormal.	Failed to modify the quota on ManageOne because the ManageOne API may be abnormal.	Check the ManageOne API status. Log in to ManageOne and go to the quota management page. If the displayed page shows the service quotas in the specified region, the ManageOne API is normal. If the ManageOne API is abnormal, try again after it restores.
400	BCS.4001753	Failed to query the product from ManageOne because the product does not exist, the product information is abnormal, or the ManageOne API is abnormal.	Failed to query the product from ManageOne because the product does not exist, the product information is abnormal, or the ManageOne API is abnormal.	Check the ManageOne API status and the product information. Log in to ManageOne and go to the quota management page. If the displayed page shows the service quotas in the specified region, the ManageOne API is normal. Then, check the BCS information. If the BCS information is missing, contact the administrator to register BCS. If the page is not displayed properly, the ManageOne API is abnormal. In this case, try again after it restores.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4001754	This operation cannot be performed on the order because the order parameters do not exist.	This operation cannot be performed on the order because the order parameters do not exist.	Buy a BCS service again.
400	BCS.4001755	Failed to save the order parameters because etcd may be abnormal.	Failed to save the order parameters because etcd may be abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4001756	The order processing cannot proceed because the order parameters have changed.	The order processing cannot proceed because the order parameters have changed.	Buy a BCS service again.
400	BCS.4001757	Operation verification failed. Only create and delete operations are supported.	Operation verification failed. Only create and delete operations are supported.	Check whether the operation is creation or deletion.
400	BCS.4001999	Failed to accept the invitation. Refresh the page and check the invitation status later.	Failed to accept the invitation. Refresh the page and check the invitation status later.	Refresh the page and check the invitation status later.
400	BCS.4003007	No authorization from CCE.	No authorization from CCE.	Obtain authorization on the CCE console.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4004001	You cannot select EVS storage because the CCE cluster version is earlier than 1.15.	You cannot select EVS storage because the CCE cluster version is earlier than 1.15.	Log in to the CCE console, choose Resource Management > Clusters , click the cluster where the BCS service is deployed, and check whether the cluster version is later than 1.15. If the cluster version is earlier than 1.15, create another cluster whose version is later than 1.15. If CCE does not support clusters whose version is later than 1.15, contact CCE support personnel.
400	BCS.4004002	EVS storage is not supported.	EVS storage is not supported.	Log in to CloudAutoDeploy-CDK, select a region, and choose Cloud Service Management > Upgrade . Select the cluster where the BCS service is deployed and select the baas-service pod for upgrade. Change the value of evs_service_enable to true , and perform the upgrade. Wait until the upgrade is complete.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4004003	You cannot select EVS storage because the EVS disk and the CCE node are located in different AZs.	You cannot select EVS storage because the EVS disk and the CCE node are located in different AZs.	Log in to the CCE console, choose Resource Management > Clusters , and click the cluster where the BCS service is deployed. Then, choose Nodes in the navigation pane, and view the AZs configured for the nodes. Choose Storage in the navigation pane, click the EVS tab, and click Buy EVS Disk . Check whether the AZ of the EVS disk is the same as that of the cluster nodes. If they are different, purchase another CCE cluster in the same AZ as the EVS disk.
400	BCS.4004004	You cannot select EVS storage when the Fabric version of the BCS service is earlier than 3.0.	You cannot select EVS storage when the Fabric version of the BCS service is earlier than 3.0.	Set the Fabric version of the BCS service to 3.0 or later.
400	BCS.4004005	You cannot change the billing mode of the BCS service to yearly/ monthly because the BCS service uses EVS storage.	You cannot change the billing mode of the BCS service to yearly/ monthly because the BCS service uses EVS storage.	BCS services billed in the yearly/ monthly mode do not support EVS storage. You can use SFS file systems instead.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4004006	Failed to query the EVS type.	Failed to query the EVS type.	Check whether the network is normal. Manually run the curl command with the EVS domain name. If the domain name cannot be accessed, contact the DNS service personnel. Log in to the EVS console, click Buy Disk , and check whether an error message is displayed. If an error message is displayed, contact the EVS O&M personnel. Check whether the back-end service of BCS is normal. If the back-end service is abnormal, view the corresponding logs.
400	BCS.400402	Query failed.	Query failed.	Check the entity type in the input parameter.
400	BCS.4004022	Failed to query the monitoring data because the AOM API may be abnormal. Check whether the AOM service is normal and whether the AOM monitoring API can be accessed.	Failed to query the monitoring data because the AOM API may be abnormal. Check whether the AOM service is normal and whether the AOM monitoring API can be accessed.	Check whether the AOM service is normal and whether the AOM monitoring API can be accessed.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4004025	Invalid request information.	Invalid request information.	Check the request information and ensure that it is correct.
400	BCS.4004029	Query failed.	Query failed.	Check the input parameters and try again.
400	BCS.4006005	Failed to query the BCS service because the etcd connection may be abnormal.	Failed to query the BCS service because the etcd connection may be abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006007	Failed to submit the request because no EIP is bound to the CCE cluster or the EIP fails to be queried.	Failed to submit the request because no EIP is bound to the CCE cluster or the EIP fails to be queried.	Log in to the CCE console. Choose Resource Management > Nodes . Locate the node corresponding to the BCS service and check whether the EIP is bound. For a private blockchain, at least one node must be bound with an EIP. For a consortium blockchain, all nodes must be bound with EIPs. Perform the following steps to bind an EIP: On the node details page, click the node name. On the displayed ECS console, click the ECS name. On the ECS details page, click the EIPs tab. Click Bind EIP . The Bind EIP dialog box is displayed.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006009	Failed to download the SDK configuration because the etcd connection or the CCE cluster status is abnormal.	Failed to download the SDK configuration because the etcd connection or the CCE cluster status is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service, and check whether the cluster status is normal and whether the cluster has available nodes. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006012	Failed to create the channel because the etcd connection or the CCE cluster status is abnormal.	Failed to create the channel because the etcd connection or the CCE cluster status is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service, and check whether the cluster status is normal and whether the cluster has available nodes. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006013	Failed to update the BCS service instance information because the etcd connection is abnormal.	Failed to update the BCS service instance information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006017	Failed to upgrade the BCS service because the etcd connection or the CCE cluster status is abnormal.	Failed to upgrade the BCS service because the etcd connection or the CCE cluster status is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service, and check whether the cluster status is normal and whether the cluster has available nodes. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006018	Failed to query data because the etcd connection is abnormal.	Failed to query data because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006019	Failed to change the password because the etcd connection, the CCE configuration update API, or the cluster status is abnormal.	Failed to change the password because the etcd connection, the CCE configuration update API, or the cluster status is abnormal.	Log in to the CCE console and check whether CCE is normal. If CCE is normal, choose Resource Management > Clusters . Locate the cluster corresponding to the BCS service and check whether the cluster status is normal. If the cluster is abnormal, try again after the cluster is restored. Alternatively, use a normal cluster to create another BCS service. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006025	Invitation failed because the tenant has already been invited or the etcd connection is abnormal.	Invitation failed because the tenant has already been invited or the etcd connection is abnormal.	Go to the Member Management page on the BCS console to check whether the tenant invitation information exists. If the invitation exists, the tenant has been invited and cannot be invited again. Wait for the invited tenant to reply. If the invitation does not exist, the etcd connection may be abnormal. In this case, proceed to the next step. Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006026	Some invitations failed because the etcd connection is abnormal.	Some invitations failed because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006027	Failed to list the members because the etcd connection is abnormal.	Failed to list the members because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006032	Failed to download the SDK configuration. The OrderEIP is empty because the etcd connection may be abnormal.	Failed to download the SDK configuration. The OrderEIP is empty because the etcd connection may be abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006034	Failed to obtain the notification information because the etcd connection is abnormal.	Failed to obtain the notification information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006036	Failed to generate the organization certificate because the etcd connection is abnormal.	Failed to generate the organization certificate because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006037	Failed to send the event information because the AOM API or the network connection is abnormal.	Failed to send the event information because the AOM API or the network connection is abnormal.	Log in to the AOM console to check whether AOM is normal and whether related information is reported. If AOM is abnormal, events can be reported only after AOM is restored. If AOM APIs are normal, the network connection may not be stable. Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4006038	Failed to update the notification information because the etcd connection is abnormal.	Failed to update the notification information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006039	Failed to submit the request because the cluster is in use.	Failed to submit the request because the cluster is in use.	Select an unused cluster and try again.
400	BCS.4006040	Failed to save the member information because the etcd connection is abnormal.	Failed to save the member information because the etcd connection is abnormal.	Rectify the etcd fault by referring to the <i>BCS Troubleshooting</i> document and try again after etcd is normal.
400	BCS.4006041	Failed to delete the notification because finished notifications cannot be deleted.	Failed to delete the notification because finished notifications cannot be deleted.	Do not delete finished notifications.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4009100	System error. Check the system or network status.	System error. Check the system or network status.	Switch from the BCS console to other service consoles, such as the CCE console. If multiple services are abnormal, the fault is caused by a common component fault in the environment. After the fault is rectified, try again. If only BCS is abnormal, the BCS backend node may be faulty. In this case, rectify the fault by following the instructions provided in "BCS Node Fault" in the <i>BCS Emergency Plan</i> .
400	BCS.4009101	The request URL does not exist. Check whether the URL has been registered.	The request URL does not exist. Check whether the URL has been registered.	Log in to the API registration page on DMK. Find BCS APIs, and check whether the API for performing the specified operation exists in the list. If it does not exist, register the API as instructed by the API Gateway document contained in the BCS installation package.

Status Code	Error Codes	Error Message	Description	Solution
400	BCS.4009102	Internal system error. Check the system status or network status.	Internal system error. Check the system status or network status.	Switch from the BCS console to other service consoles, such as the CCE console. If multiple services are abnormal, the fault is caused by a common component fault in the environment. After the fault is rectified, try again. If only BCS is abnormal, the BCS backend node may be faulty. In this case, rectify the fault by following the instructions provided in "BCS Node Fault" in the <i>BCS Emergency Plan</i> .
400	BCS.4009103	The maximum number (5) of invited members in the channel has been reached.	The maximum number (5) of invited members in the channel has been reached.	Do not exceed the maximum number of members that can be invited to each channel.
401	BCS.4010401	Permission verification failed.	Permission verification failed.	Configure the permissions by following the instructions provided in <i>BCS User Guide > Service Deployment > Prerequisites</i> .

Status Code	Error Codes	Error Message	Description	Solution
403	BCS.4030403	Insufficient permissions.	Insufficient permissions.	Configure the permissions by following the instructions provided in <i>BCS User Guide > Service Deployment > Prerequisites</i> .

7.3 Obtaining a Project ID

Obtaining a Project ID by Calling an API

You can obtain a project ID by calling an API.

The API for obtaining a project ID is **GET `https://{Endpoint}/v3/projects/`**. *{Endpoint}* indicates the endpoint of IAM. For details on API calling authentication, see [Authentication](#).

In the following example, **id** indicates a 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"
  }
}
```

Obtaining a Project ID on the Console

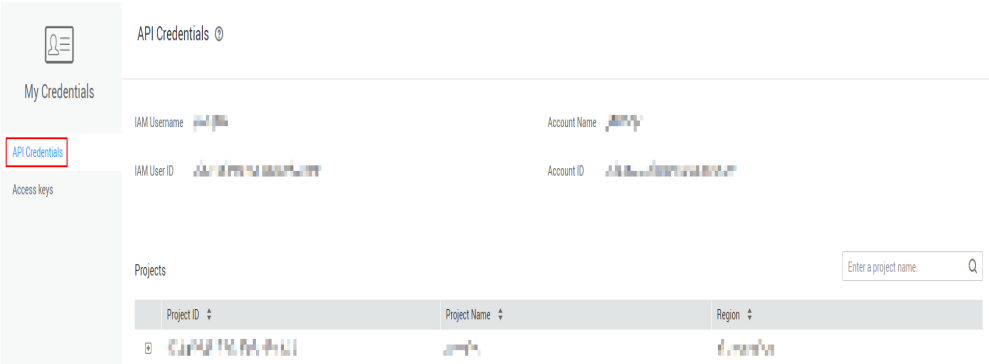
A project ID is required for some URLs when an API is called.

To obtain a project ID, perform the following operations:

Step 1 Log in to the management console.

- Step 2** Hover the mouse pointer over the username and choose **My Credentials** from the drop-down list.

Figure 7-1 Viewing project IDs



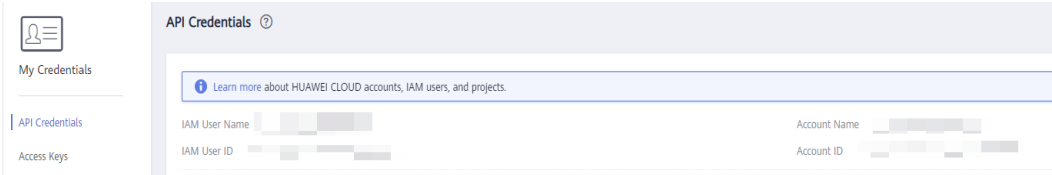
----End

7.4 Obtaining the ID

An ID (**domain-id**) is required for some URLs when an API is called. To obtain the ID, perform the following operations:

- Step 1** Log in to the management console.
- Step 2** Hover over the username and click **My Credentials** from the drop-down list.
- On the **API Credentials** page, view the **Account ID**.

Figure 7-2 Obtaining an account ID



----End

8 Change History

Released On	Description
2020-10-30	This issue is the first official release.