

Config

API Reference

Issue 01
Date 2023-12-20



Copyright © Huawei Technologies Co., Ltd. 2024. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Security Declaration

Vulnerability

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

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

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

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

Contents

1 Before You Start.....	1
1.1 Introduction.....	1
1.2 API Calling.....	1
1.3 Endpoints.....	1
1.4 Concepts.....	1
1.5 Querying Data by Page.....	3
2 API Overview.....	4
3 Calling APIs.....	6
3.1 Making an API Request.....	6
3.2 Authentication.....	10
3.3 Response.....	12
4 APIs.....	14
4.1 Resource Query.....	14
4.1.1 Querying Resources of a Specific Type.....	14
4.1.2 Listing Cloud Services.....	21
4.1.3 Querying a Resource.....	28
4.1.4 Listing Resources Recorded by the Resource Recorder.....	34
4.1.5 Querying how many resources are recorded by the resource recorder.....	41
4.1.6 Querying resource tags recorded by the resource recorder.....	47
4.1.7 Querying resource overview recorded by the resource recorder.....	52
4.1.8 Querying a specific resource recorded by the resource recorder.....	58
4.1.9 Querying All Resources Under Your Account.....	64
4.1.10 Querying a Resource Under Your Account.....	71
4.1.11 Querying Resource Tags.....	77
4.1.12 Querying the Number of Resources.....	83
4.1.13 Querying a Resource Summary.....	88
4.2 Resource Recorder.....	93
4.2.1 Querying the Resource Recorder.....	94
4.2.2 Deleting the Resource Recorder.....	100
4.2.3 Creating or Modifying the Resource Recorder.....	104
4.3 Resource Relationships.....	111
4.3.1 Querying Resource Relationships.....	111

4.3.2 Querying Details About Resource Relationships.....	117
4.4 Resource Change Records.....	124
4.4.1 Querying Change Records of a Resource.....	124
4.5 Compliance.....	133
4.5.1 Querying All Built-in Policies.....	133
4.5.2 Querying a Built-in Policy.....	143
4.5.3 Adding a Rule.....	150
4.5.4 Querying All Rules.....	159
4.5.5 Modifying a Rule.....	167
4.5.6 Querying a Rule.....	177
4.5.7 Deleting a Rule.....	184
4.5.8 Enabling a Rule.....	188
4.5.9 Disabling a Rule.....	192
4.5.10 Running a Resource Compliance Evaluation.....	197
4.5.11 Querying the Evaluation Status of a Rule.....	201
4.5.12 Querying the Compliance Results of a Resource.....	206
4.5.13 Querying All Compliance Results of Resources Evaluated by a Rule.....	212
4.5.14 Querying All Compliance Results of Resources Under a User.....	219
4.5.15 Updating the Compliance Result.....	226
4.5.16 Creating an Organization Rule.....	233
4.5.17 Querying Organization Rules.....	242
4.5.18 Querying a Specific Organization Rule.....	249
4.5.19 Deleting an Organization Rule.....	255
4.5.20 Updating an Organization Rule.....	259
4.5.21 Querying Statuses of Organization Rules.....	267
4.5.22 Querying Details About the Rule Statuses of Each Member Account in an Organization.....	274
4.6 Region Management.....	280
4.6.1 Querying Regions Visible to You.....	280
4.7 Advanced Queries.....	285
4.7.1 Running Advanced Queries.....	285
4.7.2 Creating an Advanced Query.....	293
4.7.3 Listing Advanced Queries.....	298
4.7.4 Querying an Advanced Query.....	304
4.7.5 Updating an Advanced Query.....	310
4.7.6 Deleting an Advanced Query.....	316
4.7.7 Querying Schemas.....	320
4.8 Resource Aggregators.....	326
4.8.1 Creating a Resource Aggregator.....	326
4.8.2 Querying Resource Aggregators.....	332
4.8.3 Querying a Specific Resource Aggregator.....	338
4.8.4 Querying the Status of a Specific Resource Aggregator Account.....	344
4.8.5 Updating a Resource Aggregator.....	350

4.8.6 Deleting a Resource Aggregator.....	356
4.8.7 Authorizing a Resource Aggregator Account.....	360
4.8.8 Querying Authorized Resource Aggregators Accounts.....	365
4.8.9 Deleting Authorization for a Resource Aggregator Account.....	370
4.8.10 Querying All Pending Aggregation Requests.....	374
4.8.11 Deleting Pending Authorization Requests.....	379
4.8.12 Querying the Number of Resources of a Resource Aggregator Account.....	383
4.8.13 Querying Resources of a Resource Aggregator Account.....	389
4.8.14 Querying Details About a Specific Resource in a Source Account.....	396
4.8.15 Performing an Advanced Query on a Specific Aggregator.....	403
4.8.16 Querying the Compliance Summary of One or More Source Accounts in an Aggregator.....	408
4.8.17 Querying Aggregated Rules.....	415
4.8.18 Querying Compliance Results of Aggregated Resources.....	422
4.8.19 Querying Details About a Specified Aggregated Rule.....	429
4.9 Conformance Packages.....	437
4.9.1 Querying Conformance Packages.....	437
4.9.2 Creating a Conformance Package.....	444
4.9.3 Querying a Conformance Package.....	451
4.9.4 Deleting a Conformance Package.....	457
4.9.5 Updating Conformance Packages.....	461
4.9.6 Querying the Compliance Results of Conformance Packages.....	468
4.9.7 Querying the Compliance Results of Rules in a Conformance Package.....	474
4.9.8 Querying the Details about Compliance Results of Resources Evaluated by Rules in a Conformance Package.....	480
4.9.9 Querying Scores of Conformance Packages.....	486
4.9.10 Querying Built-in Conformance Package Templates.....	492
4.9.11 Querying a Built-in Conformance Package Template.....	499
4.9.12 Create a conformance package.....	505
4.9.13 List organization conformance packages.....	511
4.9.14 View organization conformance packages.....	517
4.9.15 Delete organization conformance packages.....	523
4.9.16 Updating Organization Conformance Packages.....	527
4.9.17 View deployment status of organization conformance packages.....	533
4.9.18 View deployment status of an organization conformance package.....	539
4.10 Resource Tags.....	546
4.10.1 Querying Resources.....	546
4.10.2 Querying Resource Statistics.....	553
4.10.3 Batch Adding Resource Tags.....	558
4.10.4 Batch Deleting Resource Tags.....	561
4.10.5 Querying Resource Tags.....	564
4.10.6 Querying Resource Tags by Project.....	567
5 Permissions Policies and Supported Actions.....	572

5.1 Permissions Policies and Supported Actions.....	572
5.2 Resource List.....	573
5.3 Resource Recorder.....	576
5.4 Resource Compliance.....	576
5.5 Advanced Queries.....	580
5.6 Resource Aggregation.....	581
5.7 Conformance Packages.....	587
5.8 Resource Tag.....	591
6 Appendixes.....	592
6.1 Error Codes.....	592
6.2 Supported Services and Resource Types.....	603
6.3 Obtaining an Account ID.....	603
6.4 Status Codes.....	604
6.5 Obtaining a Project ID.....	605
7 Change History.....	607

1 Before You Start

1.1 Introduction

With Config, you can search for, record, and continuously evaluate your resource configuration to make sure that your expectations are met.

This document describes the Config APIs. Parameter description and examples are provided. You can use provided APIs to perform operations on Config. For details about all supported operations, see [API Overview](#).

If you plan to access Config through an API, ensure that you are familiar with Config concepts. For details, see [What Is Config](#).

1.2 API Calling

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

1.3 Endpoints

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

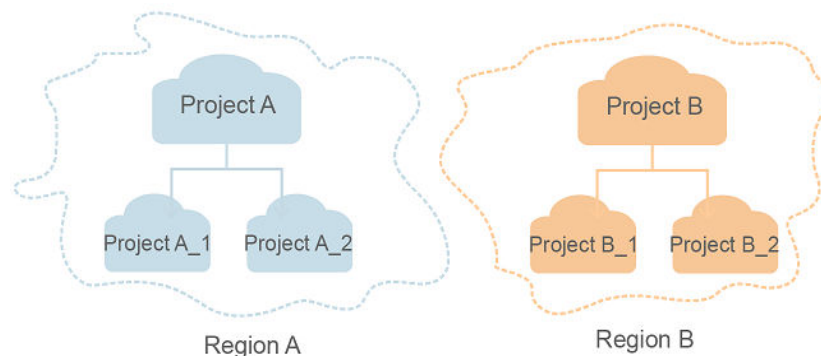
1.4 Concepts

- Organization
The Organizations service helps you manage multiple accounts within your organization. It enables you to consolidate multiple Huawei Cloud accounts into a single organization so that you can centrally manage these accounts. You can use Organizations to apply access policies to the different accounts in your organization. This helps you better meet the security and compliance requirements of your business.
- Account

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

- **User**
An IAM user is created by an account through IAM to use cloud services. Each IAM user has its own identity credentials (password and access keys). 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.
For details, see [Region and AZ](#).
- **AZ**
An AZ comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.
- **Project**
A project corresponds to a region. Default projects are defined to group and physically isolate resources (including compute, storage, and network resources) across regions. You can grant users permissions by project, so that authorized users can access all resources in the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolation model



- **Enterprise project**
Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated. An enterprise project can

contain resources of multiple regions, and resources can be added to or removed from enterprise projects.

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

1.5 Querying Data by Page

Some Config APIs support pagination query if you add **limit** and **marker** to the request URL. The value of **marker** must be the same as that returned in the last pagination query.

Table 1-1 Config parameter description

Parameter	Type	Mandatory	Description
limit	Integer	No	Restricts the number of records displayed on each page. If limit is invalid, error code 400 will be returned.
marker	String	No	Specifies the marker value returned in the last pagination query. If marker is invalid, error code 400 will be returned.

2 API Overview

Table 2-1 API description

API	Description
Resource Query	Query resources, resource tags, resource quantity and summary, resources recorded by the resource recorder, and listing cloud services.
Resource Recorder	APIs for adding, deleting, modifying, and querying the resource recorder
Resource Relationships	APIs for querying resource relationships and details
Resource Change Records	API for querying resource change records
Compliance	APIs for querying compliance policies, running rules, obtaining and updating compliance results, and managing rules and organization rules
Region Management	API for querying regions visible to a user
Advanced Queries	APIs for running, adding, deleting, modifying, and querying advanced queries
Aggregator	APIs for managing resource aggregators and authorization for resource aggregator accounts, querying resources of a resource aggregator account and running advanced queries on a resource aggregator

API	Description
Conformance Package	APIs for adding, deleting, modifying, and querying conformance packages or organization conformance packages, and for querying compliance data.
Resource Tag	APIs for listing resources, querying the number of resources, batch adding and deleting resource tags, and querying resource and project tags.

3 Calling APIs

3.1 Making an API Request

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

Request URI

A request URI is in the following format:

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

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

Table 3-1 URI parameter description

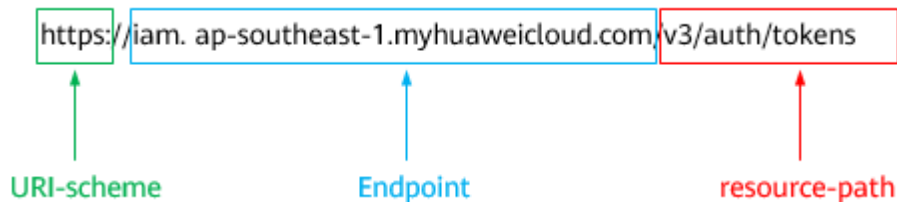
Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from Regions and Endpoints . For example, the endpoint of IAM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

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

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

`https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens`

Figure 3-1 Example URI



NOTE

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

Request Methods

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

Table 3-2 HTTP methods

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

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

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

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

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

Common request header fields are as follows.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of <i>Hostname:Port number</i> . If the port number is not specified, the default port is used. The default port number for https is 443 .	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Specifies the type (or format) of the message body. The default value application/json is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495

Parameter	Description	Mandatory	Example Value
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in Obtaining a Project ID .	No This field is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	Specifies the user token. It is a response to the API for obtaining a user token (This is the only API that does not require authentication). After the request is processed, the value of X-Subject-Token in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZlhvcNAQcCo...ggg1BBIINPXsidG9rZ

 **NOTE**

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

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

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

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

(Optional) Request Body

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

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

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

domainname, *\$ADMIN_PASS* (login password), and *xxxxxxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from [Regions and Endpoints](#).

 NOTE

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

POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "$ADMIN_PASS", //You are advised to store it in ciphertext in the
configuration file or an environment variable and decrypt it when needed to ensure security.
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

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

3.2 Authentication

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

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

Token Authentication

 NOTE

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

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for

calling the API. You can obtain a token by calling the [Obtaining a User Token](#) API.

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

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

Config is a global service. When you call the API, set **auth.scope** in the request body to **domain**. For details about how to obtain the user token, see [Obtaining a User Token](#).

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

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

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

AK/SK Authentication

NOTE

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

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

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

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

 **NOTE**

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

3.3 Response

Status Code

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

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

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

Response Header

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

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

 **NOTE**

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

Figure 3-2 Header fields of the response to the request for 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 → [REDACTED]
x-xss-protection → 1; mode=block
```

(Optional) Response Body

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

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

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

```

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

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

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

4 APIs

4.1 Resource Query

4.1.1 Querying Resources of a Specific Type

Function

This API is used to query specified resources. To call this API, you must have the **rms:resources:list** permission. For example, if you need to query the `ecs.cloudservers` resource type, set the **provider** to **ecs**, and **type** to **cloudservers** in the API request. For details about the cloud services (provider) and resource types (type), see the Supported Services and Resource Types section in the appendix.

Calling Method

For details, see [Calling APIs](#).

URI

GET `/v1/resource-manager/domains/{domain_id}/provider/{provider}/type/{type}/resources`

Table 4-1 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
provider	Yes	String	Specifies the cloud service name. Maximum: 20

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the resource type. Maximum: 32

Table 4-2 Query Parameters

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 36
ep_id	No	String	Specifies the enterprise project ID. Maximum: 36
tag	No	Map<String,String>	Specifies the tag.
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-3 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-4 Response body parameters

Parameter	Type	Description
resources	Array of ResourceEntity objects	Specifies the resource list.
page_info	PageInfo object	Specifies the pagination object.

Table 4-5 ResourceEntity

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
project_name	String	Specifies the project name in IaaS OpenStack.
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
provisioning_state	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Table 4-6 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-7 Response body parameters

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

Status code: 403

Table 4-8 Response body parameters

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

Status code: 500

Table 4-9 Response body parameters

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

Example Requests

Querying all VMs in current account

GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/provider/ecs/type/cloudServers/resources

Example Responses

Status code: 200

Operation succeeded.

```
{
  "page_info": {
    "current_count": 2,
    "next_marker": null
  },
  "resources": [ {
    "checksum": "89ca775e88e04b2c200ccbf9e219ad0d7da42e3f446e5c953d443288134eec41",
    "created": "2020-02-21T08:41:05Z",
    "ep_id": "0",
    "ep_name": "default",
    "id": "7ffd8564-d88a-4bc9-ab51-d8b79a57d0e6",
    "name": "ecs-test-1",
    "project_id": "059b5e0a2500d5552fa1c00adada8c06",
    "project_name": "project_name",
    "properties": {
      "status": "ACTIVE"
    },
    "provider": "ecs",
    "provisioning_state": "Succeeded",
    "region_id": "regionid1",
    "tags": {
      "use": "test"
    },
    "type": "cloudServers",
    "updated": "2020-02-21T08:41:05Z"
  }, {
    "checksum": "db2aad42804951c03a724b7501da9b6b4c14d319dd319d76bb7c658f256a37b0",
    "created": "2020-02-24T08:43:05Z",
    "ep_id": "0",
    "ep_name": "default",
    "id": "b63b33b7-f48c-4048-995b-0445d124a445",
    "name": "ecs-test-2",
    "project_id": "059b5e0a2500d5552fa1c00adada8c06",
    "project_name": "project_name_1",
    "properties": {
      "status": "ACTIVE"
    },
    "provider": "ecs",
    "provisioning_state": "Succeeded",
    "region_id": "regionid1",
    "tags": {
      "use": "test1"
    },
    "type": "cloudServers",
    "updated": "2020-08-11T11:55:08Z"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListResourcesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        ListResourcesRequest request = new ListResourcesRequest();
        request.withRegionId("<region_id>");
        request.withEpld("<ep_id>");
        request.withTag("<tag>");
        request.withLimit("<limit>");
        request.withMarker("<marker>");
        try {
            ListResourcesResponse response = client.listResources(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \
```

```
client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListResourcesRequest()
    request.region_id = "<region_id>"
    request.ep_id = "<ep_id>"
    request.tag = "<tag>"
    request.limit = <limit>
    request.marker = "<marker>"
    response = client.list_resources(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListResourcesRequest{}
    regionIdRequest := "<region_id>"
    request.RegionId = &regionIdRequest
    epIdRequest := "<ep_id>"
    request.Epld = &epIdRequest
    tagRequest := "<tag>"
    request.Tag = &tagRequest
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    response, err := client.ListResources(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Sever error.

Error Codes

See [Error Codes](#).

4.1.2 Listing Cloud Services

Function

Querying cloud services, resources, and regions

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/providers

Table 4-10 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-11 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Specifies the pagination offset. Minimum: 1 Maximum: 1000
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
track	No	String	Specifies whether resources are collected by default. tracked indicates that resources are collected by default, and untracked indicates that resources are not collected by default.

Request Parameters

Table 4-12 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Specifies the language of the information to return. The default name is zh-cn , which can be zh-cn and en-us . Default: zh-cn
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-13 Response body parameters

Parameter	Type	Description
resource_providers	Array of ResourceProviderResponse objects	Specifies the list of cloud service details.
total_count	Integer	Specifies the total number of cloud services.

Table 4-14 ResourceProviderResponse

Parameter	Type	Description
provider	String	Specifies the cloud service name.
display_name	String	Specifies the display name of the cloud service. You can set the language by configuring X-Language in the request header.
category_display_name	String	Specifies the display name of the cloud service type. You can set the language by configuring X-Language in the request header.
resource_types	Array of ResourceTypeResponse objects	Specifies the resource type list.

Table 4-15 ResourceTypeResponse

Parameter	Type	Description
name	String	Specifies the resource type.
display_name	String	Specifies the display name of the resource type. You can set the language by configuring X-Language in the request header.
global	Boolean	Specifies whether the resource is a global resource.
regions	Array of strings	Specifies the list of supported regions.
console_endpoint_id	String	Specifies the console endpoint ID.
console_list_url	String	Specifies the URL of the resource list page on the console.

Parameter	Type	Description
console_detail_url	String	Specifies the URL of the resource details page on the console.
track	String	Specifies whether resources are collected by default. tracked indicates that resources are collected by default, and untracked indicates that resources are not collected by default.

Status code: 500

Table 4-16 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "total_count": 2,
  "resource_providers": [ {
    "provider": "ecs",
    "display_name": "ECS",
    "category_display_name": "Compute",
    "resource_types": [ {
      "name": "cloudservers",
      "display_name": "Cloud servers",
      "global": false,
      "regions": [ "regionid1", "regionid2", "regionid3", "regionid4", "regionid5", "regionid6" ],
      "console_endpoint_id": "ecm",
      "console_list_url": "#/ecs/manager/vmList",
      "console_detail_url": "#/ecs/manager/ecsDetail?instanceId={id}",
      "track": "tracked"
    } ]
  } ], {
  "provider": "vpc",
  "display_name": "VPC",
  "category_display_name": "Networking",
  "resource_types": [ {
    "name": "vpcs",
    "display_name": "VPC",
    "global": false,
    "regions": [ "regionid1", "regionid2", "regionid3", "regionid4", "regionid5", "regionid6" ],
    "console_endpoint_id": "vpc",
    "console_list_url": "#/vpcs",

```

```
"console_detail_url" : "#/vpc/vpcmanager/vpcDetail/subnets?vpcId={id}",
"track" : "tracked"
}, {
  "name" : "bandwidths",
  "display_name" : "Shared bandwidth",
  "global" : false,
  "regions" : [ "regionid1", "regionid2", "regionid3", "regionid4", "regionid5", "regionid6" ],
  "console_endpoint_id" : "vpc",
  "console_list_url" : "#/vpc/vpcmanager/shareBandwidth",
  "console_detail_url" : "#/vpc/vpcmanager/shareBandwidth?bandwidthId={id}",
  "track" : "tracked"
}, {
  "name" : "securityGroups",
  "display_name" : "Security groups",
  "global" : false,
  "regions" : [ "regionid1", "regionid2", "regionid5", "regionid6" ],
  "console_endpoint_id" : "vpc",
  "console_list_url" : "#/secGroups",
  "console_detail_url" : "#/vpc/vpcmanager/sgDetail/sgRules?instanceId={id}",
  "track" : "tracked"
}, {
  "name" : "publicips",
  "display_name" : "EIPs",
  "global" : false,
  "regions" : [ "regionid1", "regionid2", "regionid3", "regionid4", "regionid6" ],
  "console_endpoint_id" : "vpc",
  "console_list_url" : "#/vpc/vpcmanager/eips",
  "console_detail_url" : "#/vpc/vpcmanager/eipDetailNew?eipId={id}",
  "track" : "tracked"
}
}
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListProvidersSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
```



```

        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
ListProvidersRequest request = new ListProvidersRequest();
request.withTrack(ListProvidersRequest.TrackEnum.fromValue("<track>"));
request.withOffset(<offset>);
request.withLimit(<limit>);
try {
    ListProvidersResponse response = client.listProviders(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListProvidersRequest()
        request.track = "<track>"
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_providers(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"

```

```

"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProvidersRequest{}
    trackRequest:= model.GetListProvidersRequestTrackEnum().<TRACK>
    request.Track = &trackRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListProviders(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.3 Querying a Resource

Function

This API is used to query details of a resource based on its ID. You must have the `rms:resources:get` permission. For example, to query ECSs whose resource type is `ecs.cloudservers`, you must set **provider** to `ecs` and **type** to `cloudservers` in the request. For details about the cloud services (**provider**) and resource types (**type**), see "Appendix-Supported Services and Resource Types".

Calling Method

For details, see [Calling APIs](#).

URI

GET `/v1/resource-manager/domains/{domain_id}/provider/{provider}/type/{type}/resources/{resource_id}`

Table 4-17 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
provider	Yes	String	Specifies the cloud service name. Maximum: 20
type	Yes	String	Specifies the resource type. Maximum: 32
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Request Parameters

Table 4-18 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-19 Response body parameters

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
project_name	String	Specifies the project name in IaaS OpenStack.
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
provisioning_state	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Status code: 400

Table 4-20 Response body parameters

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

Status code: 403

Table 4-21 Response body parameters

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

Status code: 404

Table 4-22 Response body parameters

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

Status code: 500

Table 4-23 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/059b5c937100d3e40ff0c00a7675a0a0/provider/ecs/type/cloudservers/resources/00337e93-82d1-40ca-911f-07cff94587cc
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "00337e93-82d1-40ca-911f-07cff94587cc",
  "name" : "dev_machine",
  "provider" : "ecs",
  "type" : "cloudservers",
  "region_id" : "regionid4",
  "project_id" : "39c2af998c334ed6bcbb75b27318f7cc",
  "project_name" : "project_name",
  "ep_id" : "0",
  "ep_name" : "default",
  "checksum" : "3a0075409edb156a74e041b7479f0d5993be1d62b4ccd2af3a1dd01ec80c8b39",
  "created" : "2019-11-20T06:24:43Z",
```

```
"updated": "2020-07-17T08:30:52Z",
"provisioning_state": "Succeeded",
"tags": {
  "usage": "Display"
},
"properties": {
  "accessIpV4": "",
  "hostName": "dev-machine",
  "addresses": [ {
    "OsExtIpsType": "fixed",
    "OsExtIpsPortId": "f2fa750a-e2ab-434f-b14a-bfe7c8cea0cc",
    "addr": "192.168.1.212",
    "version": 4,
    "OsExtIpsMacAddr": "fa:16:3e:6e:cf:33"
  }, {
    "OsExtIpsType": "floating",
    "OsExtIpsPortId": "f2fa750a-e2ab-434f-b14a-bfe7c8cea0cc",
    "addr": "100.85.225.33",
    "version": 4,
    "OsExtIpsMacAddr": "fa:16:3e:6e:cf:33"
  } ],
  "accessIpV6": "",
  "metadata": {
    "chargingMode": "0",
    "meteringImageType": "private",
    "imageName": "resource-manager-devmachine-template",
    "meteringImageId": "9bcaace4-b8da-4008-a352-3f72e1f25333",
    "meteringResourcesPerCode": "si2.large.2.linux",
    "vpclId": "cf403ef5-90df-4e7e-829d-5d21b1cb7d1e",
    "osBit": "64"
  },
  "OsExtStsVmState": "active",
  "configDrive": "",
  "OsExtStsPowerState": 1,
  "hostId": "3c381dcfc3e628c1a504ad94ba8c4e89081306455273701333f32921",
  "securityGroup": [ {
    "name": "default",
    "id": "5d55b397-ad9c-462d-af72-6599cb941c49"
  } ],
  "ExtVolumesAttached": [ {
    "bootIndex": "0",
    "id": "010d940e-a73e-417b-85ae-51b76c0d2ba0",
    "device": "/dev/vda"
  } ],
  "userId": "e311190745e94cc09d62d5779e55912d",
  "flavor": {
    "disk": "0",
    "name": "Si2.large.2",
    "id": "Si2.large.2",
    "vcpus": "2",
    "ram": "4096"
  },
  "OsDcfDiskConfig": "MANUAL",
  "hostStatus": "UP",
  "OsSrvUsgLaunchedAt": "2019-11-20T06:24:56.000000",
  "OsExtAz": "regionid4a",
  "progress": 0,
  "locked": false,
  "OS-EXT-SRV-ATTR": {
    "hostName": "dev-machine",
    "kernelId": "",
    "ramdiskId": "",
    "reservationId": "r-hhux9o7m",
    "instanceName": "instance-0009cb50",
    "host": "regionid4a-pod01.regionid4a",
    "rootDeviceName": "/dev/vda",
    "hypervisorHostName": "nova001@2",
    "launchIndex": 0
  },
}
```

```
"status" : "ACTIVE"  
}  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
public class ShowResourceByIdSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new GlobalCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ConfigClient client = ConfigClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ShowResourceByIdRequest request = new ShowResourceByIdRequest();  
        try {  
            ShowResourceByIdResponse response = client.showResourceById(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import GlobalCredentials  
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkconfig.v1 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowResourceByIdRequest()
        response = client.show_resource_by_id(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowResourceByIdRequest{}
    response, err := client.ShowResourceById(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```


More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.4 Listing Resources Recorded by the Resource Recorder

Function

Querying all resources recorded by the resource recorder. To call this API, you must have the rms:resources:list permission.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/tracked-resources

Table 4-24 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-25 Query Parameters

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 36
ep_id	No	String	Specifies the enterprise project ID. Maximum: 36
type	No	String	Specifies the resource type in the format of provider.type . Maximum: 40
limit	No	Integer	Specifies the maximum number of resources to return. Minimum: 1 Maximum: 200 Default: 100
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
id	No	String	Specifies the resource ID. Maximum: 512
name	No	String	Specifies the resource name. Maximum: 256
tags	No	Array	Specifies tags. The format is key or key=value . Array Length: 1 - 5
resource_deleted	No	Boolean	Indicating whether deleted resources need to be returned. This parameter is set to false by default. Default: false

Request Parameters

Table 4-26 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-27 Response body parameters

Parameter	Type	Description
resources	Array of ResourceEntity objects	Specifies the resource list.
page_info	PageInfo object	Specifies the pagination object.

Table 4-28 ResourceEntity

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
project_name	String	Specifies the project name in IaaS OpenStack.
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.

Parameter	Type	Description
provisioning_state	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Table 4-29 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-30** Response body parameters

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

Status code: 403**Table 4-31** Response body parameters

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

Status code: 500

Table 4-32 Response body parameters

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

Example Requests

- Querying all resources in the current account

```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/tracked-resources
```
- Querying resources in the **default** enterprise project. 100 records are returned by default.

```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/tracked-resources?limit=100&ep_id=0
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "resources": [ {
    "id": "3ccd9191-6a5e-4939-a971-4652db18b370",
    "name": "elb-265a",
    "provider": "elb",
    "type": "loadbalancers",
    "region_id": "regionid1",
    "project_id": "05498e12458025102ff5c0061a584a9f",
    "project_name": "regionid1_region_service",
    "ep_id": "0",
    "ep_name": "default",
    "checksum": "6e0271b107b764b19acb235f45c0d852f72104fe1d4b32970686e7eae8e87bf4",
    "created": "2020-02-29T09:39:19Z",
    "updated": "2020-02-29T09:39:19Z",
    "provisioning_state": "Succeeded",
    "tags": { },
    "properties": {
      "tenant_id": "05498e12458025102ff5c0061a584a9f",
      "listeners": [ {
        "id": "37de3be0-1803-43e2-9bb5-243b4b30b771"
      } ],
      "provisioning_status": "ACTIVE",
      "description": "",
      "pools": [ {
        "id": "a7ae4e7f-3e97-48ea-bdbc-334684aafac4"
      } ],
      "created_at": "2020-02-29T09:39:19"
    }
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": "CAESJhIkNGEzNWYxMmYtZjM0NC00YTnkLWE1NDEtZWVjMDBiNDIzMjg1GgTGWJss"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListTrackedResourcesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        ListTrackedResourcesRequest request = new ListTrackedResourcesRequest();
        request.withRegionId("<region_id>");
        request.withEpld("<ep_id>");
        request.withType("<type>");
        request.withLimit("<limit>");
        request.withMarker("<marker>");
        request.withId("<id>");
        request.withName("<name>");
        request.withTags();
        try {
            ListTrackedResourcesResponse response = client.listTrackedResources(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8
```

```
from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTrackedResourcesRequest()
        request.region_id = "<region_id>"
        request.ep_id = "<ep_id>"
        request.type = "<type>"
        request.limit = <limit>
        request.marker = "<marker>"
        request.id = "<id>"
        request.name = "<name>"
        request.tags =
        response = client.list_tracked_resources(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```

Build()

request := &model.ListTrackedResourcesRequest{}
regionIdRequest:= "<region_id>"
request.RegionId = &regionIdRequest
epIdRequest:= "<ep_id>"
request.EpId = &epIdRequest
typeRequest:= "<type>"
request.Type = &typeRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
idRequest:= "<id>"
request.Id = &idRequest
nameRequest:= "<name>"
request.Name = &nameRequest
response, err := client.ListTrackedResources(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
    
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.5 Querying how many resources are recorded by the resource recorder

Function

Querying how many resources are recorded by the resource recorder in the current account

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/tracked-resources/count

Table 4-33 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-34 Query Parameters

Parameter	Mandatory	Type	Description
id	No	String	Specifies the resource ID. Maximum: 512
name	No	String	Specifies the resource name. Maximum: 256
type	No	Array	Specifies resource types in the format of provider.type . Array Length: 1 - 100
region_id	No	Array	Specifies the regions. Array Length: 1 - 10
ep_id	No	Array	Specifies enterprise project IDs. Array Length: 1 - 10
project_id	No	Array	Specifies the project ID. Array Length: 1 - 10
tags	No	Array	Specifies tags. The format is key or key=value . Array Length: 1 - 5
resource_deleted	No	Boolean	Indicating whether deleted resources need to be returned. This parameter is set to false by default. Default: false

Request Parameters

Table 4-35 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-36 Response body parameters

Parameter	Type	Description
total_count	Integer	Specifies the total number of resources.

Status code: 400

Table 4-37 Response body parameters

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

Status code: 403

Table 4-38 Response body parameters

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

Status code: 500

Table 4-39 Response body parameters

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

Example Requests

Querying how many resources are recorded by the resource recorder in the current account

```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/tracked-resources/count
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "total_count" : 345
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CountTrackedResourcesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
```

```
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
CountTrackedResourcesRequest request = new CountTrackedResourcesRequest();
request.withId("<id>");
request.withName("<name>");
request.withType();
request.withRegionId();
request.withEpId();
request.withProjectId();
request.withTags();
try {
    CountTrackedResourcesResponse response = client.countTrackedResources(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CountTrackedResourcesRequest()
        request.id = "<id>"
        request.name = "<name>"
        request.type =
        request.region_id =
        request.ep_id =
        request.project_id =
        request.tags =
        response = client.count_tracked_resources(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CountTrackedResourcesRequest{}
    idRequest:= "<id>"
    request.Id = &idRequest
    nameRequest:= "<name>"
    request.Name = &nameRequest
    response, err := client.CountTrackedResources(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the [Sample Code](#) tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.6 Querying resource tags recorded by the resource recorder

Function

Querying resource tags recorded by the resource recorder in the current account

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/tracked-resources/tags

Table 4-40 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-41 Query Parameters

Parameter	Mandatory	Type	Description
key	No	String	Specifies the name of the tag key. Maximum: 128
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
limit	No	Integer	Specifies the maximum number of resources to return. Minimum: 1 Maximum: 200 Default: 100

Parameter	Mandatory	Type	Description
resource_deleted	No	Boolean	Indicating whether deleted resources need to be returned. This parameter is set to false by default. Default: false

Request Parameters

Table 4-42 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-43 Response body parameters

Parameter	Type	Description
tags	Array of TagEntity objects	Specifies tags.
page_info	PageInfo object	Specifies the pagination object.

Table 4-44 TagEntity

Parameter	Type	Description
key	String	Specifies the tag key.
value	Array of strings	Specifies tag values.

Table 4-45 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-46 Response body parameters

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

Status code: 403

Table 4-47 Response body parameters

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

Status code: 500

Table 4-48 Response body parameters

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

Example Requests

Querying resource tags recorded by the resource recorder in the current account


```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/tracked-resources/tags
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "tags" : [ {
    "key" : "chloe",
    "value" : [ "a", "b" ]
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListTrackedResourceTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListTrackedResourceTagsRequest request = new ListTrackedResourceTagsRequest();
        request.withKey("<key>");
        request.withMarker("<marker>");
        request.withLimit("<limit>");
        try {
            ListTrackedResourceTagsResponse response = client.listTrackedResourceTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
        }
    }
}
```

```
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTrackedResourceTagsRequest()
        request.key = "<key>"
        request.marker = "<marker>"
        request.limit = <limit>
        response = client.list_tracked_resource_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

```

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListTrackedResourceTagsRequest{}
keyRequest:= "<key>"
request.Key = &keyRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
response, err := client.ListTrackedResourceTags(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
    
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.7 Querying resource overview recorded by the resource recorder

Function

Querying resource overview recorded by the resource recorder in the current account

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/tracked-resources/summary

Table 4-49 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-50 Query Parameters

Parameter	Mandatory	Type	Description
name	No	String	Specifies the resource name. Maximum: 256
type	No	Array	Specifies resource types in the format of provider.type . Array Length: 1 - 100
region_id	No	Array	Specifies the regions. Array Length: 1 - 10
ep_id	No	Array	Specifies enterprise project IDs. Array Length: 1 - 10
project_id	No	Array	Specifies the project ID. Array Length: 1 - 10
tags	No	Array	Specifies tags. The format is key or key=value . Array Length: 1 - 5
resource_deleted	No	Boolean	Indicating whether deleted resources need to be returned. This parameter is set to false by default. Default: false

Request Parameters

Table 4-51 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-52 Response body parameters

Parameter	Type	Description
[items]	Array of ResourceSummaryResponseItem objects	Specifies the list of resource summary information.

Table 4-53 ResourceSummaryResponseItem

Parameter	Type	Description
provider	String	Specifies the cloud service name.
types	Array of types objects	Specifies the resource type list.

Table 4-54 types

Parameter	Type	Description
type	String	Specifies the resource type.
regions	Array of regions objects	Specifies the regions.

Table 4-55 regions

Parameter	Type	Description
region_id	String	Specifies the region ID.

Parameter	Type	Description
count	Long	Specifies the number of resources of this type in the current region.

Status code: 400

Table 4-56 Response body parameters

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

Status code: 403

Table 4-57 Response body parameters

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

Status code: 500

Table 4-58 Response body parameters

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

Example Requests

Querying resource overview recorded by the resource recorder in the current account

GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/tracked-resources/summary

Example Responses

Status code: 200

Successful Operation

```
[ {
  "provider": "ecs",
  "types": [ {
    "type": "buckets",
    "regions": [ {
      "region_id": "regionid1",
      "count": 5
    } ]
  } ]
} ]
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CollectTrackedResourcesSummarySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CollectTrackedResourcesSummaryRequest request = new CollectTrackedResourcesSummaryRequest();
        request.setName("<name>");
        request.setType();
        request.withRegionId();
        request.withEpld();
        request.withProjectId();
        request.withTags();
        try {
            CollectTrackedResourcesSummaryResponse response =
                client.collectTrackedResourcesSummary(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
        }
    }
}
```

```
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CollectTrackedResourcesSummaryRequest()
        request.name = "<name>"
        request.type =
        request.region_id =
        request.ep_id =
        request.project_id =
        request.tags =
        response = client.collect_tracked_resources_summary(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
```



```

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CollectTrackedResourcesSummaryRequest{}
nameRequest:= "<name>"
request.Name = &nameRequest
response, err := client.CollectTrackedResourcesSummary(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
    
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Successful Operation
400	Invalid Param Supplied
403	Authentication Failed
500	Internal Error

Error Codes

See [Error Codes](#).

4.1.8 Querying a specific resource recorded by the resource recorder

Function

Querying a specific resource recorded by the resource recorder in the current account

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/tracked-resources/{resource_id}

Table 4-59 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Request Parameters

Table 4-60 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-61 Response body parameters

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
project_name	String	Specifies the project name in IaaS OpenStack.

Parameter	Type	Description
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
provisioning_s tate	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Status code: 400

Table 4-62 Response body parameters

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

Status code: 403

Table 4-63 Response body parameters

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

Status code: 500

Table 4-64 Response body parameters

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

Example Requests

Querying a specific resource recorded by the resource recorder in the current account

```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/tracked-resources/{resource_id}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "00337e93-82d1-40ca-911f-07cff94587cc",
  "name" : "dev_machine",
  "provider" : "vpc",
  "type" : "vpcs",
  "region_id" : "regionid1",
  "project_id" : "39c2af998c334ed6bcbb75b27318f7cc",
  "project_name" : "regionid1",
  "ep_id" : "0",
  "ep_name" : "default",
  "checksum" : "3a0075409edb156a74e041b7479f0d5993be1d62b4ccd2af3a1dd01ec80c8b39",
  "created" : "2019-11-20T06:24:43Z",
  "updated" : "2020-07-17T08:30:52Z",
  "provisioning_state" : "Succeeded",
  "tags" : {
    "usage" : "mm1"
  },
  "properties" : {
    "cidr" : "192.168.0.0/16",
    "status" : "OK"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;
```

```
public class ShowTrackedResourceDetailSolution {
    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowTrackedResourceDetailRequest request = new ShowTrackedResourceDetailRequest();
        try {
            ShowTrackedResourceDetailResponse response = client.showTrackedResourceDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowTrackedResourceDetailRequest()
        response = client.show_tracked_resource_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowTrackedResourceDetailRequest{}
    response, err := client.ShowTrackedResourceDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.9 Querying All Resources Under Your Account

Function

This API is used to query all resources under your account and you must have the `**rms:resources:list` permission.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/all-resources

Table 4-65 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-66 Query Parameters

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 36
ep_id	No	String	Specifies the enterprise project ID. Maximum: 36
type	No	String	Specifies the resource type in the format of provider.type . Maximum: 40
limit	No	Integer	Specifies the maximum number of resources to return. Minimum: 1 Maximum: 200 Default: 100

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
id	No	String	Specifies the resource ID. Maximum: 512
name	No	String	Specifies the resource name. Maximum: 256
tags	No	Array	Specifies tags. The format is key or key=value . Array Length: 1 - 5

Request Parameters

Table 4-67 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-68 Response body parameters

Parameter	Type	Description
resources	Array of ResourceEntity objects	Specifies the resource list.
page_info	PageInfo object	Specifies the pagination object.

Table 4-69 ResourceEntity

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
project_name	String	Specifies the project name in IaaS OpenStack.
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
provisioning_s tate	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Table 4-70 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-71 Response body parameters

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

Status code: 403

Table 4-72 Response body parameters

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

Status code: 500

Table 4-73 Response body parameters

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

Example Requests

- Querying all resources under your account
GET `https://{endpoint}/v1/resource-manager/domains/{domain_id}/all-resources`
- Querying your resources in the **default** enterprise project and setting to return the first 100 records
GET `https://{endpoint}/v1/resource-manager/domains/{domain_id}/all-resources?limit=100&ep_id=0`

Example Responses

Status code: 200

Operation succeeded.

```
{
  "resources": [ {
    "id": "3ccd9191-6a5e-4939-a971-4652db18b370",
    "name": "elb-265a",
    "provider": "elb",
    "type": "loadbalancers",
    "region_id": "regionid1",
```

```
"project_id" : "05498e12458025102ff5c0061a584a9f",
"project_name" : "regionid1_region_service",
"ep_id" : "0",
"ep_name" : "default",
"checksum" : "6e0271b107b764b19acb235f45c0d852f72104fe1d4b32970686e7eae8e87bf4",
"created" : "2020-02-29T09:39:19Z",
"updated" : "2020-02-29T09:39:19Z",
"provisioning_state" : "Succeeded",
"tags" : { },
"properties" : {
  "tenant_id" : "05498e12458025102ff5c0061a584a9f",
  "listeners" : [ {
    "id" : "37de3be0-1803-43e2-9bb5-243b4b30b771"
  } ],
  "provisioning_status" : "ACTIVE",
  "description" : ""
}
}, {
  "id" : "a6e56d05501944d3b2507ba506a43744",
  "name" : "console",
  "provider" : "cdn",
  "type" : "domains",
  "region_id" : "global",
  "project_id" : "",
  "project_name" : "",
  "ep_id" : "0",
  "ep_name" : "default",
  "checksum" : "56afa8b76428f90e9abf5cbf33535d8816166114d32eeb119658d6c59eceda",
  "created" : "2020-01-04T13:42:37Z",
  "updated" : "2020-01-15T04:23:01Z",
  "provisioning_state" : "Succeeded",
  "tags" : { },
  "properties" : {
    "domain_name" : "console",
    "domain_status" : "offline",
    "business_type" : "WEB",
    "modify_time" : 1579062181463,
    "cname" : "console"
  }
} ],
"page_info" : {
  "current_count" : 2,
  "next_marker" : null
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListAllResourcesSolution {
```

```

public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");

    ICredential auth = new GlobalCredentials()
        .withAk(ak)
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ListAllResourcesRequest request = new ListAllResourcesRequest();
    request.withRegionId("<region_id>");
    request.withEpld("<ep_id>");
    request.withType("<type>");
    request.withLimit(<limit>);
    request.withMarker("<marker>");
    request.withId("<id>");
    request.withName("<name>");
    request.withTags();
    try {
        ListAllResourcesResponse response = client.listAllResources(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:

```

```
request = ListAllResourcesRequest()
request.region_id = "<region_id>"
request.ep_id = "<ep_id>"
request.type = "<type>"
request.limit = <limit>
request.marker = "<marker>"
request.id = "<id>"
request.name = "<name>"
request.tags =
response = client.list_all_resources(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAllResourcesRequest{}
    regionIdRequest:= "<region_id>"
    request.RegionId = &regionIdRequest
    epIdRequest:= "<ep_id>"
    request.EpId = &epIdRequest
    typeRequest:= "<type>"
    request.Type = &typeRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    idRequest:= "<id>"
    request.Id = &idRequest
    nameRequest:= "<name>"
    request.Name = &nameRequest
    response, err := client.ListAllResources(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User Authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.10 Querying a Resource Under Your Account

Function

This API is used to query a resource under your account.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/all-resources/{resource_id}

Table 4-74 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Request Parameters

Table 4-75 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-76 Response body parameters

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
project_name	String	Specifies the project name in IaaS OpenStack.
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
provisioning_state	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.

Parameter	Type	Description
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Status code: 400

Table 4-77 Response body parameters

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

Status code: 403

Table 4-78 Response body parameters

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

Status code: 500

Table 4-79 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/059b5c937100d3e40ff0c00a7675a0a0/all-resources/00337e93-82d1-40ca-911f-07cff94587cc
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "00337e93-82d1-40ca-911f-07cff94587cc",
  "name" : "dev_machine",
```



```

"provider" : "ecs",
"type" : "cloudservers",
"region_id" : "regionid4",
"project_id" : "39c2af998c334ed6bcb75b27318f7cc",
"project_name" : "project_name",
"ep_id" : "0",
"ep_name" : "chloe",
"checksum" : "3a0075409edb156a74e041b7479f0d5993be1d62b4ccd2af3a1dd01ec80c8b39",
"created" : "2019-11-20T06:24:43Z",
"updated" : "2020-07-17T08:30:52Z",
"provisioning_state" : "Succeeded",
"tags" : {
  "usage" : "Display"
},
"properties" : {
  "accessIpv4" : "",
  "hostName" : "dev-machine",
  "addresses" : [ {
    "OsExtIpsType" : "fixed",
    "OsExtIpsPortId" : "f2fa750a-e2ab-434f-b14a-bfe7c8cea0cc",
    "addr" : "192.168.1.212",
    "version" : 4,
    "OsExtIpsMacAddr" : "fa:16:3e:6e:cf:33"
  }, {
    "OsExtIpsType" : "floating",
    "OsExtIpsPortId" : "f2fa750a-e2ab-434f-b14a-bfe7c8cea0cc",
    "addr" : "100.85.225.33",
    "version" : 4,
    "OsExtIpsMacAddr" : "fa:16:3e:6e:cf:33"
  } ],
  "accessIpv6" : "",
  "metadata" : {
    "chargingMode" : "0",
    "meteringImageType" : "private",
    "imageName" : "resource-manager-devmachine-template",
    "meteringImageId" : "9bcaace4-b8da-4008-a352-3f72e1f25333",
    "meteringResourcesPerCode" : "si2.large.2.linux",
    "vpclId" : "cf403ef5-90df-4e7e-829d-5d21b1cb7d1e",
    "osBit" : "64"
  },
  "OsExtStsVmState" : "active",
  "configDrive" : "",
  "OsExtStsPowerState" : 1,
  "hostId" : "3c381dcfc3e628c1a504ad94ba8c4e89081306455273701333f32921",
  "securityGroup" : [ {
    "name" : "default",
    "id" : "5d55b397-ad9c-462d-af72-6599cb941c49"
  } ],
  "ExtVolumesAttached" : [ {
    "bootIndex" : "0",
    "id" : "010d940e-a73e-417b-85ae-51b76c0d2ba0",
    "device" : "/dev/vda"
  } ],
  "userId" : "e311190745e94cc09d62d5779e55912d",
  "flavor" : {
    "disk" : "0",
    "name" : "Si2.large.2",
    "id" : "Si2.large.2",
    "vcpus" : "2",
    "ram" : "4096"
  },
  "OsDcfDiskConfig" : "MANUAL",
  "hostStatus" : "UP",
  "OsSrvUsgLaunchedAt" : "2019-11-20T06:24:56.000000",
  "OsExtAz" : "regionid4a",
  "progress" : 0,
  "locked" : false,
  "OS-EXT-SRV-ATTR" : {
    "hostName" : "dev-machine",

```

```
"kernelId" : "",
"ramdiskId" : "",
"reservationId" : "r-hhux9o7m",
"instanceName" : "instance-0009cb50",
"host" : "regionid4a-pod01.regionid4",
"rootDeviceName" : "/dev/vda",
"hypervisorHostName" : "nova001@2",
"launchIndex" : 0
},
"status" : "ACTIVE"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowResourceDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowResourceDetailRequest request = new ShowResourceDetailRequest();
        try {
            ShowResourceDetailResponse response = client.showResourceDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowResourceDetailRequest()
        response = client.show_resource_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowResourceDetailRequest{}
    response, err := client.ShowResourceDetail(request)
```

```
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.11 Querying Resource Tags

Function

This API is used to query all resource tags under your account.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/all-resources/tags

Table 4-80 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-81 Query Parameters

Parameter	Mandatory	Type	Description
key	No	String	Specifies the name of the tag key. Maximum: 128
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
limit	No	Integer	Specifies the maximum number of resources to return. Minimum: 1 Maximum: 200 Default: 100

Request Parameters

Table 4-82 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-83 Response body parameters

Parameter	Type	Description
tags	Array of TagEntity objects	Specifies tags.
page_info	PageInfo object	Specifies the pagination object.

Table 4-84 TagEntity

Parameter	Type	Description
key	String	Specifies the tag key.
value	Array of strings	Specifies tag values.

Table 4-85 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-86 Response body parameters

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

Status code: 403

Table 4-87 Response body parameters

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

Status code: 500

Table 4-88 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/059b5c937100d3e40ff0c00a7675a0a0/all-resources/tags
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "tags": [ {
    "key": "chloe",
    "value": [ "a", "b" ]
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListAllTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ListAllTagsRequest request = new ListAllTagsRequest();
request.withKey("<key>");
request.withMarker("<marker>");
request.withLimit(<limit>);
try {
    ListAllTagsResponse response = client.listAllTags(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAllTagsRequest()
        request.key = "<key>"
        request.marker = "<marker>"
        request.limit = <limit>
        response = client.list_all_tags(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```



```

"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAllTagsRequest{}
    keyRequest:= "<key>"
    request.Key = &keyRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListAllTags(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.12 Querying the Number of Resources

Function

This API is used to query the number of resources.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/all-resources/count

Table 4-89 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-90 Query Parameters

Parameter	Mandatory	Type	Description
id	No	String	Specifies the resource ID. Maximum: 512
name	No	String	Specifies the resource name. Maximum: 256
type	No	Array	Specifies resource types in the format of provider.type . Array Length: 1 - 100
region_id	No	Array	Specifies the regions. Array Length: 1 - 10
ep_id	No	Array	Specifies enterprise project IDs. Array Length: 1 - 10
project_id	No	Array	Specifies the project ID. Array Length: 1 - 10
tags	No	Array	Specifies tags. The format is key or key=value . Array Length: 1 - 5

Request Parameters

Table 4-91 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-92 Response body parameters

Parameter	Type	Description
total_count	Integer	Specifies the total number of resources.

Status code: 400

Table 4-93 Response body parameters

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

Status code: 403

Table 4-94 Response body parameters

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

Status code: 500

Table 4-95 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/059b5c937100d3e40ff0c00a7675a0a0/all-resources/count
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "total_count" : 345
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CountAllResourcesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CountAllResourcesRequest request = new CountAllResourcesRequest();
```

```
request.withId("<id>");
request.withName("<name>");
request.withType();
request.withRegionId();
request.withEpId();
request.withProjectId();
request.withTags();
try {
    CountAllResourcesResponse response = client.countAllResources(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CountAllResourcesRequest()
        request.id = "<id>"
        request.name = "<name>"
        request.type =
        request.region_id =
        request.ep_id =
        request.project_id =
        request.tags =
        response = client.count_all_resources(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CountAllResourcesRequest{}
    idRequest:= "<id>"
    request.Id = &idRequest
    nameRequest:= "<name>"
    request.Name = &nameRequest
    response, err := client.CountAllResources(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.1.13 Querying a Resource Summary

Function

This API is used to query a resource summary.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/all-resources/summary

Table 4-96 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-97 Query Parameters

Parameter	Mandatory	Type	Description
name	No	String	Specifies the resource name. Maximum: 256
type	No	Array	Specifies resource types in the format of provider.type . Array Length: 1 - 100
region_id	No	Array	Specifies the regions. Array Length: 1 - 10
ep_id	No	Array	Specifies enterprise project IDs. Array Length: 1 - 10
project_id	No	Array	Specifies the project ID. Array Length: 1 - 10
tags	No	Array	Specifies tags. The format is key or key=value . Array Length: 1 - 5

Request Parameters

Table 4-98 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-99 Response body parameters

Parameter	Type	Description
[items]	Array of ResourceSummaryResponseItem objects	Specifies the list of resource summary information.

Table 4-100 ResourceSummaryResponseItem

Parameter	Type	Description
provider	String	Specifies the cloud service name.
types	Array of types objects	Specifies the resource type list.

Table 4-101 types

Parameter	Type	Description
type	String	Specifies the resource type.
regions	Array of regions objects	Specifies the regions.

Table 4-102 regions

Parameter	Type	Description
region_id	String	Specifies the region ID.
count	Long	Specifies the number of resources of this type in the current region.

Status code: 400

Table 4-103 Response body parameters

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

Status code: 403

Table 4-104 Response body parameters

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

Status code: 500

Table 4-105 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/059b5c937100d3e40ff0c00a7675a0a0/all-resources/summary
```

Example Responses

Status code: 200

Operation succeeded.

```
[ {
  "provider": "obs",
  "types": [ {
    "type": "buckets",
    "regions": [ {
      "region_id": "regionid1",
      "count": 5
    } ]
  } ]
}, {
  "provider": "ecs",
  "types": [ {
    "type": "cloudservers",
    "regions": [ {
      "region_id": "regionid2",
      "count": 1
    } ]
  } ]
} ]
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CollectAllResourcesSummarySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        CollectAllResourcesSummaryRequest request = new CollectAllResourcesSummaryRequest();
        request.setName("<name>");
        request.withType();
        request.withRegionId();
        request.withEpId();
        request.withProjectId();
        request.withTags();
        try {
            CollectAllResourcesSummaryResponse response = client.collectAllResourcesSummary(request);
```

```
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CollectAllResourcesSummaryRequest()
        request.name = "<name>"
        request.type =
        request.region_id =
        request.ep_id =
        request.project_id =
        request.tags =
        response = client.collect_all_resources_summary(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CollectAllResourcesSummaryRequest{}
nameRequest := "<name>"
request.Name = &nameRequest
response, err := client.CollectAllResourcesSummary(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.2 Resource Recorder

4.2.1 Querying the Resource Recorder

Function

This API is used to query details about the resource recorder.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/tracker-config

Table 4-106 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-107 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-108 Response body parameters

Parameter	Type	Description
channel	ChannelConfigBody object	Specifies configurations for the tracker channel.
selector	SelectorConfigBody object	Specifies the selector.

Parameter	Type	Description
retention_period_in_days	Integer	Number of days for data storage.
agency_name	String	Specifies the IAM agency name.

Table 4-109 ChannelConfigBody

Parameter	Type	Description
smn	TrackerSMNChannelConfigBody object	Specifies configurations for the SMN channel. For details about how to grant other accounts the permissions for publishing messages to SMN topics, see Cross-Account Authorization (Resource Recorder > Enabling, Configuring, or Modifying the Resource Recorder) in the <i>User Guide</i> .
obs	TrackerOBSCChannelConfigBody object	Specifies configurations for the OBS bucket. For details about how to grant other accounts the permissions for dumping files to OBS buckets, see Cross-Account Authorization (Resource Recorder > Enabling, Configuring, or Modifying the Resource Recorder) in the <i>User Guide</i> .

Table 4-110 TrackerSMNChannelConfigBody

Parameter	Type	Description
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID.
topic_urn	String	Specifies the SMN topic URN.

Table 4-111 TrackerOBSCChannelConfigBody

Parameter	Type	Description
bucket_name	String	Specifies the OBS bucket name. Maximum: 63
bucket_prefix	String	OBS bucket prefix Maximum: 256
region_id	String	Specifies the region ID.

Table 4-112 SelectorConfigBody

Parameter	Type	Description
all_supported	Boolean	Specifies whether to select all supported resources.
resource_types	Array of strings	Specifies the resource type list.

Status code: 400

Table 4-113 Response body parameters

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

Status code: 403

Table 4-114 Response body parameters

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

Status code: 404

Table 4-115 Response body parameters

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

Status code: 500

Table 4-116 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "agency_name" : "rms_tracker_agency",
  "channel" : {
    "obs" : {
      "bucket_name" : "rmtest",
      "region_id" : "regionid1"
    },
    "smn" : {
      "project_id" : "059b5e0a2500d5552fa1c00adada8c06",
      "region_id" : "regionid1",
      "topic_urn" : "urn:smn:regionid1:059b5e0a2500d5552fa1c00adada8c06:agencytest"
    }
  },
  "selector" : {
    "all_supported" : true,
    "resource_types" : [ ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowTrackerConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```



```
ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowTrackerConfigRequest request = new ShowTrackerConfigRequest();
try {
    ShowTrackerConfigResponse response = client.showTrackerConfig(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowTrackerConfigRequest()
        response = client.show_tracker_config(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
```

```

"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowTrackerConfigRequest{}
    response, err := client.ShowTrackerConfig(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the [Sample Code](#) tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No resource recorder found.
500	Server error.

Error Codes

See [Error Codes](#).

4.2.2 Deleting the Resource Recorder

Function

This API is used to delete the resource recorder.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/tracker-config

Table 4-117 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-118 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-119 Response body parameters

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

Status code: 403**Table 4-120** Response body parameters

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

Status code: 404**Table 4-121** Response body parameters

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

Status code: 500**Table 4-122** Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeleteTrackerConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteTrackerConfigRequest request = new DeleteTrackerConfigRequest();
        try {
            DeleteTrackerConfigResponse response = client.deleteTrackerConfig(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteTrackerConfigRequest()
```

```
response = client.delete_tracker_config(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteTrackerConfigRequest{}
    response, err := client.DeleteTrackerConfig(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.

Status Code	Description
404	No resource recorder found.
500	Server error.

Error Codes

See [Error Codes](#).

4.2.3 Creating or Modifying the Resource Recorder

Function

This API is used to create or modify the resource recorder. Only one resource recorder can be configured.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/tracker-config

Table 4-123 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-124 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-125 Request body parameters

Parameter	Mandatory	Type	Description
channel	Yes	ChannelConfigBody object	Specifies configurations for the tracker channel.
selector	Yes	SelectorConfigBody object	Specifies the selector.
retention_period_in_days	No	Integer	Number of days for data storage.
agency_name	Yes	String	Specifies the IAM agency name.

Table 4-126 ChannelConfigBody

Parameter	Mandatory	Type	Description
smn	No	TrackerSMNChannelConfigBody object	Specifies configurations for the SMN channel. For details about how to grant other accounts the permissions for publishing messages to SMN topics, see Cross-Account Authorization (Resource Recorder > Enabling, Configuring, or Modifying the Resource Recorder) in the <i>User Guide</i> .
obs	No	TrackerOBSCChannelConfigBody object	Specifies configurations for the OBS bucket. For details about how to grant other accounts the permissions for dumping files to OBS buckets, see Cross-Account Authorization (Resource Recorder > Enabling, Configuring, or Modifying the Resource Recorder) in the <i>User Guide</i> .

Table 4-127 TrackerSMNChannelConfigBody

Parameter	Mandatory	Type	Description
region_id	Yes	String	Specifies the region ID.
project_id	Yes	String	Specifies the project ID.
topic_urn	Yes	String	Specifies the SMN topic URN.

Table 4-128 TrackerOBSChannelConfigBody

Parameter	Mandatory	Type	Description
bucket_name	Yes	String	Specifies the OBS bucket name. Maximum: 63
bucket_prefix	No	String	OBS bucket prefix Maximum: 256
region_id	Yes	String	Specifies the region ID.

Table 4-129 SelectorConfigBody

Parameter	Mandatory	Type	Description
all_supported	Yes	Boolean	Specifies whether to select all supported resources.
resource_types	Yes	Array of strings	Specifies the resource type list.

Response Parameters

Status code: 400

Table 4-130 Response body parameters

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

Status code: 403

Table 4-131 Response body parameters

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

Status code: 404

Table 4-132 Response body parameters

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

Status code: 500

Table 4-133 Response body parameters

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

Example Requests

Creating the resource recorder and enabling message pushing and dumping

PUT /v1/resource-manager/domains/{domain_id}/tracker-config

```
{
  "channel" : {
    "smn" : {
      "region_id" : "regionid1",
      "project_id" : "39c2af998c334ed6bcbb75b27318f7b5",
      "topic_urn" : "urn:smn:regionid1:39c2af998c334ed6bcbb75b27318f7b5:resource-manager-test"
    },
    "obs" : {
      "bucket_name" : "config-snapshot",
      "region_id" : "regionid1"
    }
  },
  "selector" : {
    "all_supported" : true,
    "resource_types" : [ ]
  },
  "agency_name" : "rms_tracker_agency"
}
```

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

Creating the resource recorder and enabling message pushing and dumping

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateTrackerConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateTrackerConfigRequest request = new CreateTrackerConfigRequest();
        TrackerConfigBody body = new TrackerConfigBody();
        SelectorConfigBody selectorbody = new SelectorConfigBody();
        selectorbody.withAllSupported(true);
        TrackerOBSChannelConfigBody obsChannel = new TrackerOBSChannelConfigBody();
        obsChannel.withBucketName("config-snapshot")
            .withRegionId("regionid1");
        TrackerSMNChannelConfigBody smnChannel = new TrackerSMNChannelConfigBody();
        smnChannel.withRegionId("regionid1")
            .withProjectId("39c2af998c334ed6bcbb75b27318f7b5")
            .withTopicUrn("urn:smn:regionid1:39c2af998c334ed6bcbb75b27318f7b5:resource-manager-test");
        ChannelConfigBody channelbody = new ChannelConfigBody();
        channelbody.withSmn(smnChannel)
            .withObs(obsChannel);
        body.withAgencyName("rms_tracker_agency");
        body.withSelector(selectorbody);
        body.withChannel(channelbody);
        request.withBody(body);
        try {
            CreateTrackerConfigResponse response = client.createTrackerConfig(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Creating the resource recorder and enabling message pushing and dumping

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateTrackerConfigRequest()
        selectorbody = SelectorConfigBody(
            all_supported=True
        )
        obsChannel = TrackerOBSChannelConfigBody(
            bucket_name="config-snapshot",
            region_id="regionid1"
        )
        smnChannel = TrackerSMNChannelConfigBody(
            region_id="regionid1",
            project_id="39c2af998c334ed6bcbb75b27318f7b5",
            topic_urn="urn:smn:regionid1:39c2af998c334ed6bcbb75b27318f7b5:resource-manager-test"
        )
        channelbody = ChannelConfigBody(
            smn=smnChannel,
            obs=obsChannel
        )
        request.body = TrackerConfigBody(
            agency_name="rms_tracker_agency",
            selector=selectorbody,
            channel=channelbody
        )
        response = client.create_tracker_config(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Creating the resource recorder and enabling message pushing and dumping

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
```

```

"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateTrackerConfigRequest{}
    selectorbody := &model.SelectorConfigBody{
        AllSupported: true,
    }
    obsChannel := &model.TrackerObsChannelConfigBody{
        BucketName: "config-snapshot",
        RegionId: "regionid1",
    }
    smnChannel := &model.TrackerSmnChannelConfigBody{
        RegionId: "regionid1",
        ProjectId: "39c2af998c334ed6bcbb75b27318f7b5",
        TopicUrn: "urn:smn:regionid1:39c2af998c334ed6bcbb75b27318f7b5:resource-manager-test",
    }
    channelbody := &model.ChannelConfigBody{
        Smn: smnChannel,
        Obs: obsChannel,
    }
    request.Body = &model.TrackerConfigBody{
        AgencyName: "rms_tracker_agency",
        Selector: selectorbody,
        Channel: channelbody,
    }
    response, err := client.CreateTrackerConfig(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.

Status Code	Description
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.3 Resource Relationships

4.3.1 Querying Resource Relationships

Function

This API is used to query the relationship between the resource with a specific ID and other resources. You can set **direction** to **in** or **out**. Resource relationships depend on enabling resource recorder. Resource relationships depend on enabling resource recorder.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/relations

Table 4-134 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Table 4-135 Query Parameters

Parameter	Mandatory	Type	Description
direction	Yes	String	Specifies the resource relationship direction.
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 1000
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-136 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-137 Response body parameters

Parameter	Type	Description
relations	Array of ResourceRelation objects	Specifies the list of the resource relationships.
page_info	PageInfo object	Specifies the pagination object.

Table 4-138 ResourceRelation

Parameter	Type	Description
relation_type	String	Specifies the relationship type.
from_resource_type	String	Specifies the type of the source resource.
to_resource_type	String	Specifies the type of the destination resource.
from_resource_id	String	Specifies the ID of the source resource.
to_resource_id	String	Specifies the ID of the destination resource.

Table 4-139 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-140 Response body parameters

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

Status code: 403

Table 4-141 Response body parameters

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

Status code: 404

Table 4-142 Response body parameters

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

Status code: 500

Table 4-143 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/resources/{resource_id}/relations?
direction=out&limit=1000
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "relations" : [ {
    "relation_type" : "isAttachedTo",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "evs.volumes",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "0075ed19-59dd-49be-961d-117bb6bfd3e"
  }, {
    "relation_type" : "contains",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "vpc.publicips",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "3813d6d3-ef88-47b1-b343-cdf6390c6dcb"
  }, {
    "relation_type" : "isAssociatedWith",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "vpc.securityGroups",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "8cca3002-00af-4812-a853-b7a6fbee06a4"
  }, {
    "relation_type" : "isAttachedTo",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "evs.volumes",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
```

```
"to_resource_id" : "f4a107eb-4c6d-4dc8-88d8-de337923956f"
}, {
  "relation_type" : "isContainedIn",
  "from_resource_type" : "ecs.cloudservers",
  "to_resource_type" : "vpc.vpcs",
  "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
  "to_resource_id" : "ff13d70d-17e5-4ec8-945a-c874e0db99d3"
}],
"page_info" : {
  "current_count" : 5,
  "next_marker" : null
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowResourceRelationsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowResourceRelationsRequest request = new ShowResourceRelationsRequest();
        request.withDirection(ShowResourceRelationsRequest.DirectionEnum.fromValue("<direction>"));
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ShowResourceRelationsResponse response = client.showResourceRelations(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import GlobalCredentials  
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkconfig.v1 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = GlobalCredentials(ak, sk) \  
  
    client = ConfigClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ShowResourceRelationsRequest()  
        request.direction = "<direction>"  
        request.limit = <limit>  
        request.marker = "<marker>"  
        response = client.show_resource_relations(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"  
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := global.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := config.NewConfigClient(  

```

```

config.ConfigClientBuilder().
    WithRegion(region.ValueOf("<YOUR REGION>")).
    WithCredential(auth).
    Build())

request := &model.ShowResourceRelationsRequest{}
request.Direction = model.GetShowResourceRelationsRequestDirectionEnum().<DIRECTION>
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
response, err := client.ShowResourceRelations(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.3.2 Querying Details About Resource Relationships

Function

This API is used to query the relationship between a resource and other resources by the resource ID. The relationship direction can be **in** or **out**. An IAM user needs to have the **rms:resources:getRelation** permissions to call this API. Resource relationships depend on enabling resource recorder.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/all-resources/{resource_id}/relations

Table 4-144 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Table 4-145 Query Parameters

Parameter	Mandatory	Type	Description
direction	Yes	String	Specifies the direction of a resource relationship.
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 1000
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-146 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-147 Response body parameters

Parameter	Type	Description
relations	Array of ResourceRelation objects	Specifies the list of the resource relationships.
page_info	PageInfo object	Specifies the pagination object.

Table 4-148 ResourceRelation

Parameter	Type	Description
relation_type	String	Specifies the relationship type.
from_resource_type	String	Specifies the type of the source resource.
to_resource_type	String	Specifies the type of the destination resource.
from_resource_id	String	Specifies the ID of the source resource.
to_resource_id	String	Specifies the ID of the destination resource.

Table 4-149 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-150 Response body parameters

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

Status code: 403

Table 4-151 Response body parameters

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

Status code: 404

Table 4-152 Response body parameters

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

Status code: 500

Table 4-153 Response body parameters

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

Example Requests

```
GET https://{endpoint}/v1/resource-manager/domains/{domain_id}/all-resources/{resource_id}/relations?
direction=out
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "relations" : [ {
    "relation_type" : "isAttachedTo",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "evs.volumes",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "0075ed19-59dd-49be-961d-117bb6bfd3e"
  }, {
    "relation_type" : "contains",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "vpc.publicips",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "3813d6d3-ef88-47b1-b343-cdf6390c6dcb"
  }, {
    "relation_type" : "isAssociatedWith",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "vpc.securityGroups",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "8cca3002-00af-4812-a853-b7a6fbee06a4"
  }, {
    "relation_type" : "isAttachedTo",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "evs.volumes",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "f4a107eb-4c6d-4dc8-88d8-de337923956f"
  }, {
    "relation_type" : "isContainedIn",
    "from_resource_type" : "ecs.cloudservers",
    "to_resource_type" : "vpc.vpcs",
    "from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
    "to_resource_id" : "ff13d70d-17e5-4ec8-945a-c874e0db99d3"
  } ],
  "page_info" : {
    "current_count" : 5,
    "next_marker" : null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowResourceRelationsDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
```



```
.withAk(ak)
.withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowResourceRelationsDetailRequest request = new ShowResourceRelationsDetailRequest();
request.withDirection(ShowResourceRelationsDetailRequest.DirectionEnum.fromValue("<direction>"));
request.withLimit(<limit>);
request.withMarker("<marker>");
try {
    ShowResourceRelationsDetailResponse response = client.showResourceRelationsDetail(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowResourceRelationsDetailRequest()
        request.direction = "<direction>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.show_resource_relations_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowResourceRelationsDetailRequest{}
    request.Direction = model.GetShowResourceRelationsDetailRequestDirectionEnum().<DIRECTION>
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    response, err := client.ShowResourceRelationsDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.4 Resource Change Records

4.4.1 Querying Change Records of a Resource

Function

This API is used to query change records of a resource and its relationships with other resources.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/history

Table 4-154 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Table 4-155 Query Parameters

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200

Parameter	Mandatory	Type	Description
earlier_time	No	Long	Specifies the start time of the query. If this parameter is not set, the earliest time is used by default.
later_time	No	Long	Specifies the end time of the query. If this parameter is not set, the current time is used by default.
chronological_order	No	String	Specifies the time sequence of the data to be returned. The default value is Reverse .

Request Parameters

Table 4-156 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-157 Response body parameters

Parameter	Type	Description
items	Array of HistoryItem objects	Specifies the list of resource history.
page_info	PageInfo object	Specifies the pagination object.

Table 4-158 HistoryItem

Parameter	Type	Description
domain_id	String	Specifies the user ID.
resource_id	String	Specifies the resource ID.
resource_type	String	Specifies the resource type.
capture_time	String	Specifies the time when the resource was captured.
status	String	Specifies the resource status.
relations	Array of ResourceRelation objects	Specifies the list of the resource relationships.
resource	ResourceEntity object	Specifies the resource object.

Table 4-159 ResourceRelation

Parameter	Type	Description
relation_type	String	Specifies the relationship type.
from_resource_type	String	Specifies the type of the source resource.
to_resource_type	String	Specifies the type of the destination resource.
from_resource_id	String	Specifies the ID of the source resource.
to_resource_id	String	Specifies the ID of the destination resource.

Table 4-160 ResourceEntity

Parameter	Type	Description
id	String	Specifies the resource ID.
name	String	Specifies the resource name.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.

Parameter	Type	Description
project_name	String	Specifies the project name in IaaS OpenStack.
ep_id	String	Specifies the enterprise project ID.
ep_name	String	Specifies the name of an enterprise project.
checksum	String	Specifies the resource checksum.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
provisioning_state	String	Specifies the status of a resource operation.
state	String	Resource state. The value can be normal or deleted.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies the detailed properties of the resource.

Table 4-161 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-162 Response body parameters

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

Status code: 403

Table 4-163 Response body parameters

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

Status code: 500

Table 4-164 Response body parameters

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

Example Requests

```
GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/history?
earlier_time=1595865600000&later_time=1603875761000&limit=10
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "items": [ {
    "domain_id": "9548dff51b6849fab3d7353d631158ee",
    "resource_id": "6af96128-d58d-426c-91e0-b38144c0f112",
    "resource_type": "ecs.cloudservers",
    "capture_time": "2020-10-12T07:44:40.167Z",
    "status": "ResourceChanged.UPDATE",
    "relations": [ {
      "relation_type": "contains",
      "from_resource_type": "ecs.cloudservers",
      "to_resource_type": "vpc.publicips",
      "from_resource_id": "6af96128-d58d-426c-91e0-b38144c0f112",
      "to_resource_id": "3813d6d3-ef88-47b1-b343-cdf6390c6dcb"
    }, {
      "relation_type": "isAttachedTo",
      "from_resource_type": "ecs.cloudservers",
      "to_resource_type": "evs.volumes",
      "from_resource_id": "6af96128-d58d-426c-91e0-b38144c0f112",
      "to_resource_id": "0075ed19-59dd-49be-961d-117bb6fbfd3e"
    }, {
      "relation_type": "isAssociatedWith",
      "from_resource_type": "ecs.cloudservers",
      "to_resource_type": "vpc.securityGroups",
      "from_resource_id": "6af96128-d58d-426c-91e0-b38144c0f112",
      "to_resource_id": "8cca3002-00af-4812-a853-b7a6fbee06a4"
    }, {
      "relation_type": "isAttachedTo",
```

```

"from_resource_type" : "ecs.cloudservers",
"to_resource_type" : "evs.volumes",
"from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
"to_resource_id" : "f4a107eb-4c6d-4dc8-88d8-de337923956f"
}, {
"relation_type" : "isContainedIn",
"from_resource_type" : "ecs.cloudservers",
"to_resource_type" : "vpc.vpcs",
"from_resource_id" : "6af96128-d58d-426c-91e0-b38144c0f112",
"to_resource_id" : "ff13d70d-17e5-4ec8-945a-c874e0db99d3"
}
],
"resource" : {
"id" : "6af96128-d58d-426c-91e0-b38144c0f112",
"name" : "david-test-cce-49831-bh9ax",
"provider" : "ecs",
"type" : "cloudservers",
"region_id" : "regionid1",
"project_id" : "15caceb7599f4ee3a770dbb70ad69b2b",
"project_name" : "project_name",
"ep_id" : "0",
"ep_name" : "default",
"checksum" : "30482e7e4f2fc6e9842689c5d498cd13862def3c7369e9216af87c2198557b28",
"created" : "2020-10-12T07:44:10Z",
"updated" : "2020-10-12T07:44:38Z",
"provisioning_state" : "Succeeded",
"tags" : {
"CCE-Dynamic-Provisioning-Node" : "80fbd7d5-0c5e-11eb-bc0e-0255ac101e85"
},
"properties" : {
"accessIpsV4" : "",
"hostName" : "david-test-cce-49831-bh9ax",
"addresses" : [ {
"OsExtIpsType" : "fixed",
"OsExtIpsPortId" : "a7bfdd6f-3a56-4bbb-b37e-1b9ccac1ac96",
"addr" : "192.168.238.230",
"version" : 4,
"OsExtIpsMacAddr" : "fa:16:3e:99:ec:b1"
}
],
"accessIpsV6" : "",
"metadata" : {
"chargingMode" : "0",
"meteringImageType" : "gold",
"imageName" : "CCE_images_EulerOS-Node-CCE20.7.1.B003-v1.17.9-r0-amd64-
f8389e5c3d65dc7b5bf8f10924d2fc16e35588fc-20200808144721",
"meteringImageId" : "e8b32b27-a103-4a1f-b955-5d6be46ae9da",
"meteringResourcesPerCode" : "si3.large.2.linux",
"vpclId" : "ff13d70d-17e5-4ec8-945a-c874e0db99d3",
"osBit" : "64"
},
"OsExtStsVmState" : "active",
"configDrive" : "",
"OsExtStsPowerState" : 1,
"hostId" : "7f007da24f79bc37782d3440e3d4faa374c0eb555c1dfc98b687c996",
"securityGroup" : [ {
"name" : "david-test-cce-cce-node-8x31o",
"id" : "8cca3002-00af-4812-a853-b7a6fbee06a4"
}
],
"ExtVolumesAttached" : [ {
"bootIndex" : "0",
"id" : "f4a107eb-4c6d-4dc8-88d8-de337923956f",
"device" : "/dev/vda"
}, {
"bootIndex" : "-1",
"id" : "0075ed19-59dd-49be-961d-117bb6bfd3e",
"device" : "/dev/vdb"
}
],
"userId" : "07729a673f8026624f3ec006f3b3cda9",
"flavor" : {
"disk" : "0",

```



```
"name" : "Si3.large.2",
"id" : "Si3.large.2",
"vcpus" : "2",
"ram" : "4096"
},
"OsDcfDiskConfig" : "MANUAL",
"hostStatus" : "UP",
"OsSrvUsgLaunchedAt" : "2020-10-12T07:44:23.000000",
"OsExtAz" : "regionid1a",
"progress" : 0,
"locked" : false,
"OS-EXT-SRV-ATTR" : {
  "hostName" : "david-test-cce-49831-bh9ax",
  "kernelId" : "",
  "ramdiskId" : "",
  "reservationId" : "r-l8y8evjo",
  "instanceName" : "instance-000b2c2d",
  "host" : "regionid1a-pod01.regionid1",
  "rootDeviceName" : "/dev/vda",
  "hypervisorHostName" : "nova001@83",
  "launchIndex" : 0
},
"status" : "ACTIVE"
}
}],
"page_info" : {
  "current_count" : 1,
  "next_marker" :
"CAESJjE2MDI0ODg2Nzc1NDYtNWY4NDA5NjU4YmI5MDUxZDBhMWEzMmJjGgQCXWaw"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowResourceHistorySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
```

```
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
        ShowResourceHistoryRequest request = new ShowResourceHistoryRequest();
        request.withEarlierTime(<earlier_time>L);
        request.withLaterTime(<later_time>L);

        request.withChronologicalOrder(ShowResourceHistoryRequest.ChronologicalOrderEnum.fromValue("<chronological_order>"));
        request.withMarker("<marker>");
        request.withLimit(<limit>);
        try {
            ShowResourceHistoryResponse response = client.showResourceHistory(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowResourceHistoryRequest()
        request.earlier_time = <earlier_time>
        request.later_time = <later_time>
        request.chronological_order = "<chronological_order>"
        request.marker = "<marker>"
        request.limit = <limit>
        response = client.show_resource_history(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowResourceHistoryRequest{
        earlierTimeRequest:= int64(<earlier_time>)
        request.EarlierTime = &earlierTimeRequest
        laterTimeRequest:= int64(<later_time>)
        request.LaterTime = &laterTimeRequest
        chronologicalOrderRequest:=
        model.GetShowResourceHistoryRequestChronologicalOrderEnum().<CHRONOLOGICAL_ORDER>
        request.ChronologicalOrder = &chronologicalOrderRequest
        markerRequest:= "<marker>"
        request.Marker = &markerRequest
        limitRequest:= int32(<limit>)
        request.Limit = &limitRequest
    }
    response, err := client.ShowResourceHistory(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.

Status Code	Description
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.5 Compliance

4.5.1 Querying All Built-in Policies

Function

This API is used to query all built-in policies.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/policy-definitions

Request Parameters

Table 4-165 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Specifies the language of the information to return. The default name is zh-cn , which can be zh-cn and en-us . Default: zh-cn
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-166 Response body parameters

Parameter	Type	Description
value	Array of PolicyDefinition objects	Specifies the policy definition list.
page_info	PageInfo object	Specifies the pagination object.

Table 4-167 PolicyDefinition

Parameter	Type	Description
id	String	Specifies the policy ID.
name	String	Specifies the policy name.
display_name	String	Specifies the policy display name.
policy_type	String	Specifies the policy type.
description	String	Specifies the description of the policy definition.
policy_rule_type	String	Specifies the syntax type of the policy.
policy_rule	Object	Specifies the policy rule.
trigger_type	String	Specifies the trigger type.
keywords	Array of strings	Specifies keywords.
default_resource_types	Array of default_resource_types objects	Specifies the list of default resource types.
parameters	Map<String, PolicyParameterDefinition >	Specifies the policy parameter.

Table 4-168 default_resource_types

Parameter	Type	Description
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.

Table 4-169 PolicyParameterDefinition

Parameter	Type	Description
name	String	Specifies the name of the policy parameter.
description	String	Specifies the description of the policy parameter.
allowed_values	Array of objects	Specifies the allowed values of the policy parameter.
default_value	String	Specifies the default value of the policy parameter.
minimum	Float	Specifies the minimum value of the policy parameter.
maximum	Float	Specifies the maximum value of the policy parameter.
min_items	Integer	Specifies the minimum number of the policy parameter.
max_items	Integer	Specifies the maximum number of the policy parameter.
min_length	Integer	Specifies the minimum string length for policy parameters or for each item.
max_length	Integer	Specifies the maximum string length for policy parameters or for each item.
pattern	String	Specifies the pattern for policy parameters or for each item.
type	String	Specifies the type of the policy parameter.

Table 4-170 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-171 Response body parameters

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

Status code: 403

Table 4-172 Response body parameters

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

Status code: 404

Table 4-173 Response body parameters

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

Status code: 500

Table 4-174 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "id": "5fa365476eed194ccb2c04d1",
    "name": "volumes-encrypted-check",
    "policy_type": "builtin",
    "description": "An EVS disk is non-compliant if it has been mounted but not encrypted.",
    "policy_rule_type": "dsl",
    "policy_rule": {
      "allOf": [ {
        "value": "${resource().provider}",
        "comparator": "equals",
        "pattern": "evs"
      }, {
        "value": "${resource().type}",
        "comparator": "equals",
        "pattern": "volumes"
      }, {
        "value": "${resource().properties.status}",
        "comparator": "equals",
        "pattern": "in-use"
      }, {
        "anyOf": [ {
          "value": "${resource().properties.metadata}",
          "comparator": "notContainsKey",
          "pattern": "systemEncrypted"
        }, {
          "value": "${resource().properties.metadata.systemEncrypted}",
          "comparator": "equals",
          "pattern": "0"
        }
      ]
    }
  } ],
  "keywords": [ "evs", "ecs" ],
  "parameters": { }
}, {
  "id": "5fa9f89b6eed194ccb2c04db",
  "name": "required-tag-check",
  "policy_type": "builtin",
  "description": "A resource is non-compliant if it does not have a specified tag.",
  "policy_rule_type": "dsl",
  "policy_rule": {
    "anyOf": [ {
      "value": "${contains(resource().tags, parameters('specifiedTagKey'))}",
      "comparator": "equals",
      "pattern": false
    }
  ]
}, {
```



```

    "allOf" : [ {
      "value" : "${length(parameters('specifiedTagValue'))}",
      "comparator" : "greater",
      "pattern" : 0
    }, {
      "value" : "${getValue(resource().tags, parameters('specifiedTagKey'))}",
      "comparator" : "notIn",
      "pattern" : "${parameters('specifiedTagValue')}}"
    } ]
  } ]
}, {
  "keywords" : [ "tag" ],
  "parameters" : {
    "specifiedTagKey" : {
      "name" : null,
      "description" : "the specified tag key.",
      "allowed_values" : null,
      "default_value" : null,
      "type" : "String"
    },
    "specifiedTagValue" : {
      "name" : null,
      "description" : "The list of allowed tag value, permit all if empty.",
      "allowed_values" : null,
      "default_value" : null,
      "type" : "Array"
    }
  }
}, {
  "id" : "5fa265c0aa1e6afc05a0ff07",
  "name" : "allowed-images-by-id",
  "policy_type" : "builtin",
  "description" : "An ECS is non-compliant if its image ID is not in the specified image ID list.",
  "policy_rule_type" : "dsl",
  "policy_rule" : {
    "allOf" : [ {
      "value" : "${resource().provider}",
      "comparator" : "equals",
      "pattern" : "ecs"
    }, {
      "value" : "${resource().type}",
      "comparator" : "equals",
      "pattern" : "cloudservers"
    }, {
      "value" : "${resource().properties.metadata.meteringImageId}",
      "comparator" : "notIn",
      "pattern" : "${parameters('listOfAllowedImages')}}"
    } ]
  },
  "keywords" : [ "ecs", "ims" ],
  "parameters" : {
    "listOfAllowedImages" : {
      "name" : null,
      "description" : "The list of allowed image IDs.",
      "allowed_values" : null,
      "default_value" : null,
      "type" : "Array"
    }
  }
}, {
  "id" : "5fa36558aa1e6afc05a3d0a7",
  "name" : "volume-inuse-check",
  "policy_type" : "builtin",
  "description" : "An EVS disk is non-compliant if it is not mounted to a cloud server.",
  "policy_rule_type" : "dsl",
  "policy_rule" : {
    "allOf" : [ {
      "value" : "${resource().provider}",
      "comparator" : "equals",

```

```

    "pattern": "evs"
  }, {
    "value": "${resource().type}",
    "comparator": "equals",
    "pattern": "volumes"
  }, {
    "value": "${resource().properties.status}",
    "comparator": "equals",
    "pattern": "available"
  }
]
},
"keywords": [ "evs" ],
"parameters": { }
}, {
  "id": "5fa3a1196eed194ccb2c04d7",
  "name": "allowed_volume_specs",
  "policy_type": "builtin",
  "description": "An EVS disk is non-compliant if its type is not in the specified disk type list.",
  "policy_rule_type": "dsl",
  "policy_rule": {
    "allOf": [ {
      "value": "${resource().provider}",
      "comparator": "equals",
      "pattern": "evs"
    }, {
      "value": "${resource().type}",
      "comparator": "equals",
      "pattern": "volumes"
    }, {
      "value": "${resource().properties.volumeType}",
      "comparator": "notIn",
      "pattern": "${parameters('listOfAllowedSpecs')}"
    }
  ]
},
"keywords": [ "evs" ],
"parameters": {
  "listOfAllowedSpecs": {
    "name": null,
    "description": "The list of allowed disk specifications.",
    "allowed_values": null,
    "default_value": null,
    "type": "Array"
  }
}
}, {
  "id": "5f8d549bffeccc14f1fb522a",
  "name": "allowed_ecs_flavors",
  "policy_type": "builtin",
  "description": "An ECS is non-compliant if its flavor is not in the specified flavor list.",
  "policy_rule_type": "dsl",
  "policy_rule": {
    "allOf": [ {
      "value": "${resource().provider}",
      "comparator": "equals",
      "pattern": "ecs"
    }, {
      "value": "${resource().type}",
      "comparator": "equals",
      "pattern": "cloudservers"
    }, {
      "value": "${resource().properties.flavor.name}",
      "comparator": "notIn",
      "pattern": "${parameters('listOfAllowedFlavors')}"
    }
  ]
},
"keywords": [ "ecs" ],
"parameters": {
  "listOfAllowedFlavors": {
    "name": null,

```

```

    "description": "The list of allowed flavor types.",
    "allowed_values": null,
    "default_value": null,
    "type": "Array"
  }
}
}, {
  "id": "5fa3a1196eed194ccb2c04d5",
  "name": "eip-unbound-check",
  "policy_type": "builtin",
  "description": "An EIP is non-compliant if it is not bound to any resources.",
  "policy_rule_type": "dsl",
  "policy_rule": {
    "allOf": [ {
      "value": "${resource().provider}",
      "comparator": "equals",
      "pattern": "vpc"
    }, {
      "value": "${resource().type}",
      "comparator": "equals",
      "pattern": "publicips"
    }, {
      "value": "${resource().properties.status}",
      "comparator": "equals",
      "pattern": "DOWN"
    }
  ]
},
  "keywords": [ "vpc" ],
  "parameters": { }
}, {
  "id": "5f8d5428ffecc14f1fb5205",
  "name": "ecs-instance-no-public-ip",
  "policy_type": "builtin",
  "description": "An ECS is non-compliant if it has an EIP.",
  "policy_rule_type": "dsl",
  "policy_rule": {
    "allOf": [ {
      "value": "${resource().provider}",
      "comparator": "equals",
      "pattern": "ecs"
    }, {
      "value": "${resource().type}",
      "comparator": "equals",
      "pattern": "cloudservers"
    }, {
      "value": "${length(filter(resource().properties.addresses, lambda x: equals(x.OsExtIpsType,
'floating'))))}",
      "comparator": "greater",
      "pattern": 0
    }
  ]
},
  "keywords": [ "ecs" ],
  "parameters": { }
}, {
  "id": "5fa265c0aa1e6afc05a0ff10",
  "name": "ecs-instance-in-vpc",
  "policy_type": "builtin",
  "description": "An ECS is non-compliant if it does not belong to a specified VPC.",
  "policy_rule_type": "dsl",
  "policy_rule": {
    "allOf": [ {
      "value": "${resource().provider}",
      "comparator": "equals",
      "pattern": "ecs"
    }, {
      "value": "${resource().type}",
      "comparator": "equals",
      "pattern": "cloudservers"
    }
  ],
  "parameters": { }
}, {

```

```
        "value" : "${resource().properties.metadata.vpcId}",
        "comparator" : "notEquals",
        "pattern" : "${parameters('vpcId')}}"
    } ]
},
"keywords" : [ "ecs", "vpc" ],
"parameters" : {
    "vpcId" : {
        "name" : null,
        "description" : "VPC ID that contains the ECS instance.",
        "allowed_values" : null,
        "default_value" : null,
        "type" : "String"
    }
}
} ],
"page_info" : {
    "current_count" : 9,
    "next_marker" : null
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListBuiltInPolicyDefinitionsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListBuiltInPolicyDefinitionsRequest request = new ListBuiltInPolicyDefinitionsRequest();
        try {
            ListBuiltInPolicyDefinitionsResponse response = client.listBuiltInPolicyDefinitions(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```

        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListBuiltInPolicyDefinitionsRequest()
        response = client.list_built_in_policy_definitions(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

```

```

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListBuiltInPolicyDefinitionsRequest{}
response, err := client.ListBuiltInPolicyDefinitions(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	Authentication failed or you do not have the operation permissions.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.2 Querying a Built-in Policy

Function

This API is used to query a built-in policy based on the policy ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/policy-definitions/{policy_definition_id}

Table 4-175 Path Parameters

Parameter	Mandatory	Type	Description
policy_definition_id	Yes	String	Specifies the policy ID. Maximum: 36

Request Parameters

Table 4-176 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Language	No	String	Specifies the language of the information to return. The default name is zh-cn , which can be zh-cn and en-us . Default: zh-cn
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-177 Response body parameters

Parameter	Type	Description
id	String	Specifies the policy ID.
name	String	Specifies the policy name.
display_name	String	Specifies the policy display name.
policy_type	String	Specifies the policy type.
description	String	Specifies the description of the policy definition.
policy_rule_type	String	Specifies the syntax type of the policy.
policy_rule	Object	Specifies the policy rule.

Parameter	Type	Description
trigger_type	String	Specifies the trigger type.
keywords	Array of strings	Specifies keywords.
default_resource_types	Array of default_resource_types objects	Specifies the list of default resource types.
parameters	Map<String, PolicyParameterDefinition >	Specifies the policy parameter.

Table 4-178 default_resource_types

Parameter	Type	Description
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.

Table 4-179 PolicyParameterDefinition

Parameter	Type	Description
name	String	Specifies the name of the policy parameter.
description	String	Specifies the description of the policy parameter.
allowed_values	Array of objects	Specifies the allowed values of the policy parameter.
default_value	String	Specifies the default value of the policy parameter.
minimum	Float	Specifies the minimum value of the policy parameter.
maximum	Float	Specifies the maximum value of the policy parameter.
min_items	Integer	Specifies the minimum number of the policy parameter.
max_items	Integer	Specifies the maximum number of the policy parameter.
min_length	Integer	Specifies the minimum string length for policy parameters or for each item.

Parameter	Type	Description
max_length	Integer	Specifies the maximum string length for policy parameters or for each item.
pattern	String	Specifies the pattern for policy parameters or for each item.
type	String	Specifies the type of the policy parameter.

Status code: 400**Table 4-180** Response body parameters

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

Status code: 403**Table 4-181** Response body parameters

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

Status code: 404**Table 4-182** Response body parameters

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

Status code: 500

Table 4-183 Response body parameters

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

Example Requests

GET https://{endpoint}/v1/resource-manager/policy-definitions/5f8d5428ffecc14f1fb5205

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "5f8d5428ffecc14f1fb5205",
  "name" : "ecs-instance-no-public-ip",
  "policy_type" : "builtin",
  "description" : "An ECS is non-compliant if it has an EIP.",
  "policy_rule_type" : "dsl",
  "policy_rule" : {
    "allOf" : [ {
      "value" : "${resource().provider}",
      "comparator" : "equals",
      "pattern" : "ecs"
    }, {
      "value" : "${resource().type}",
      "comparator" : "equals",
      "pattern" : "cloudservers"
    }, {
      "value" : "${length(filter(resource().properties.addresses, lambda x: equals(x.OsExtIpsType, 'floating')))}",
      "comparator" : "greater",
      "pattern" : 0
    }
  ]
},
  "keywords" : [ "ecs" ],
  "parameters" : { }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowBuiltInPolicyDefinitionSolution {
```

```

public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");

    ICredential auth = new GlobalCredentials()
        .withAk(ak)
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ShowBuiltInPolicyDefinitionRequest request = new ShowBuiltInPolicyDefinitionRequest();
    try {
        ShowBuiltInPolicyDefinitionResponse response = client.showBuiltInPolicyDefinition(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowBuiltInPolicyDefinitionRequest()
        response = client.show_built_in_policy_definition(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowBuiltInPolicyDefinitionRequest{}
    response, err := client.ShowBuiltInPolicyDefinition(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	Policy not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.3 Adding a Rule

Function

This API is used to add a rule.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/policy-assignments

Table 4-184 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-185 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-186 Request body parameters

Parameter	Mandatory	Type	Description
policy_assignment_type	No	String	Specifies the rule type, which can be builtin or custom .
name	Yes	String	Specifies the rule name. Maximum: 64

Parameter	Mandatory	Type	Description
description	No	String	Specifies the rule description. Maximum: 512
period	No	String	Specifies the trigger period.
policy_filter	No	PolicyFilterDefinition object	Specifies the policy filter of a rule.
policy_definition_id	No	String	Specifies the policy definition ID. Maximum: 36
custom_policy	No	CustomPolicy object	Specifies the custom rule.
parameters	No	Map<String, PolicyParameterValue >	Specifies the rule parameters.
tags	No	Array of ResourceTag objects	

Table 4-187 PolicyFilterDefinition

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 128
resource_provider	No	String	Specifies the cloud service name. Maximum: 128
resource_type	No	String	Specifies the resource type. Maximum: 128
resource_id	No	String	Specifies the resource ID. Maximum: 512
tag_key	No	String	Specifies the tag key. Maximum: 128
tag_value	No	String	Specifies the tag value. Maximum: 256

Table 4-188 CustomPolicy

Parameter	Mandatory	Type	Description
function_urn	Yes	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	Yes	String	Specifies how function is called by a custom rule.
auth_value	No	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-189 PolicyParameterValue

Parameter	Mandatory	Type	Description
value	No	Object	Specifies the value of the rule parameter.

Table 4-190 ResourceTag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	Yes	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Response Parameters

Status code: 200

Table 4-191 Response body parameters

Parameter	Type	Description
policy_assignment_type	String	Specifies the rule type, which can be builtin or custom .
id	String	Specifies the rule ID.
name	String	Specifies the rule name.
description	String	Specifies the rule description.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
period	String	Specifies how often the rule is triggered, which can be One_Hour , Three_Hours , Six_Hours , Twelve_Hours , or TwentyFour_Hours .
state	String	Specifies the rule status.
created	String	Specifies the time when the rule was added.
updated	String	Specifies the time when the rule was modified.
policy_definition_id	String	Specifies the ID of the policy used by the rule.
custom_policy	CustomPolicy object	Specifies the custom rule.
parameters	Map<String, PolicyParameter >	Specifies the rule parameters.
tags	Array of ResourceTag objects	
created_by	String	Specifies the person who created the rule.

Table 4-192 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128

Parameter	Type	Description
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-193 CustomPolicy

Parameter	Type	Description
function_urn	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	String	Specifies how function is called by a custom rule.
auth_value	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-194 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Table 4-195 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Status code: 400

Table 4-196 Response body parameters

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

Status code: 403

Table 4-197 Response body parameters

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

Status code: 500

Table 4-198 Response body parameters

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

Example Requests

Create a rule named "allowed-images-by-id".

```
PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/policy-assignments
{
  "name": "allowed-images-by-id",
  "description": "The ECS resource is non-compliant if the image it used is not in the allowed list",
  "parameters": {
    "listOfAllowedImages": {
      "value": [ "ea0d6e0e-99c3-406d-a873-3bb45462b624" ]
    }
  },
  "policy_filter": {
    "resource_provider": "ecs",
    "resource_type": "cloudservers"
  },
  "policy_definition_id": "5f3c8b9a9811a13739a32ed9"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "5fb76a240f2daf0b9662a04d",
  "name" : "volume-inuse-check",
  "description" : "An EVS disk is non-compliant if it is not mounted to a cloud server.",
  "policy_filter" : {
    "region_id" : null,
    "resource_provider" : "evs",
    "resource_type" : "volumes",
    "resource_id" : null,
    "tag_key" : null,
    "tag_value" : null
  },
  "period" : null,
  "state" : "Enabled",
  "created" : "2020-11-20T07:03:00",
  "updated" : "2020-11-20T07:03:00",
  "policy_definition_id" : "5fa36558aa1e6afc05a3d0a7",
  "parameters" : { }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Create a rule named "allowed-images-by-id".

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.Map;
import java.util.HashMap;

public class CreatePolicyAssignmentsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
    }
}
```

```
CreatePolicyAssignmentsRequest request = new CreatePolicyAssignmentsRequest();
PolicyAssignmentRequestBody body = new PolicyAssignmentRequestBody();
PolicyParameterValue parametersParameter = new PolicyParameterValue();
parametersParameters.addValue("ea0d6e0e-99c3-406d-a873-3bb45462b624");
Map<String, PolicyParameterValue> listbodyParameters = new HashMap<>();
listbodyParameters.put("listOfAllowedImages", parametersParameter);
PolicyFilterDefinition policyFilterbody = new PolicyFilterDefinition();
policyFilterbody.withResourceProvider("ecs")
    .withResourceType("cloudservers");
body.withParameters(listbodyParameters);
body.withPolicyDefinitionId("5f3c8b9a9811a13739a32ed9");
body.withPolicyFilter(policyFilterbody);
body.withDescription("The ECS resource is non-compliant if the image it used is not in the allowed
list");
body.withName("allowed-images-by-id");
request.withBody(body);
try {
    CreatePolicyAssignmentsResponse response = client.createPolicyAssignments(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Create a rule named "allowed-images-by-id".

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreatePolicyAssignmentsRequest()
        parametersParameter = PolicyParameterValue(
            value="ea0d6e0e-99c3-406d-a873-3bb45462b624"
        )
        listParametersbody = {
            "listOfAllowedImages": parametersParameter
        }
        policyFilterbody = PolicyFilterDefinition(
```

```
        resource_provider="ecs",
        resource_type="cloudservers"
    )
    request.body = PolicyAssignmentRequestBody(
        parameters=listParametersbody,
        policy_definition_id="5f3c8b9a9811a13739a32ed9",
        policy_filter=policyFilterbody,
        description="The ECS resource is non-compliant if the image it used is not in the allowed list",
        name="allowed-images-by-id"
    )
    response = client.create_policy_assignments(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Create a rule named "allowed-images-by-id".

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreatePolicyAssignmentsRequest{}
    var valueParameters interface{} = "[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
    parametersParameters := model.PolicyParameterValue{
        Value: &valueParameters,
    }
    var listParametersbody = map[string](model.PolicyParameterValue){
        "listOfAllowedImages": parametersParameters,
    }
    resourceProviderPolicyFilter:= "ecs"
    resourceTypePolicyFilter:= "cloudservers"
    policyFilterbody := &model.PolicyFilterDefinition{
        ResourceProvider: &resourceProviderPolicyFilter,
        ResourceType: &resourceTypePolicyFilter,
    }
    policyDefinitionIdPolicyAssignmentRequestBody:= "5f3c8b9a9811a13739a32ed9"
    descriptionPolicyAssignmentRequestBody:= "The ECS resource is non-compliant if the image it used is
    not in the allowed list"
```

```

request.Body = &model.PolicyAssignmentRequestBody{
    Parameters: listParametersbody,
    PolicyDefinitionId: &policyDefinitionIdPolicyAssignmentRequestBody,
    PolicyFilter: policyFilterbody,
    Description: &descriptionPolicyAssignmentRequestBody,
    Name: "allowed-images-by-id",
}
response, err := client.CreatePolicyAssignments(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.4 Querying All Rules

Function

This API is used to query all rules.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/policy-assignments

Table 4-199 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-200 Query Parameters

Parameter	Mandatory	Type	Description
policy_assignment_name	No	String	Specifies the assignment name. Maximum: 64
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 500 Default: 500
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-201 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-202 Response body parameters

Parameter	Type	Description
value	Array of PolicyAssignment objects	Specifies rules.
page_info	PageInfo object	Specifies the pagination object.

Table 4-203 PolicyAssignment

Parameter	Type	Description
policy_assignment_type	String	Specifies the rule type, which can be builtin or custom .
id	String	Specifies the rule ID.
name	String	Specifies the rule name.
description	String	Specifies the rule description.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
period	String	Specifies how often the rule is triggered, which can be One_Hour , Three_Hours , Six_Hours , Twelve_Hours , or TwentyFour_Hours .
state	String	Specifies the rule status.
created	String	Specifies the time when the rule was added.
updated	String	Specifies the time when the rule was modified.
policy_definition_id	String	Specifies the ID of the policy used by the rule.
custom_policy	CustomPolicy object	Specifies the custom rule.
parameters	Map<String, PolicyParameterValue >	Specifies the rule parameters.
tags	Array of ResourceTag objects	
created_by	String	Specifies the person who created the rule.

Table 4-204 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-205 CustomPolicy

Parameter	Type	Description
function_urn	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	String	Specifies how function is called by a custom rule.
auth_value	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-206 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Table 4-207 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Table 4-208 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-209** Response body parameters

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

Status code: 403**Table 4-210** Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 500

Table 4-211 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "id": "5fb76a240f2daf0b9662a04d",
    "name": "volume-inuse-check",
    "description": "An EVS disk is non-compliant if it is not mounted to a cloud server.",
    "policy_filter": {
      "region_id": null,
      "resource_provider": "evs",
      "resource_type": "volumes",
      "resource_id": null,
      "tag_key": null,
      "tag_value": null
    },
    "state": "Enabled",
    "created": "2020-11-20T07:03:00",
    "updated": "2020-11-20T07:03:00",
    "policy_definition_id": "5fa36558aa1e6afc05a3d0a7",
    "parameters": { }
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListPolicyAssignmentsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPolicyAssignmentsRequest request = new ListPolicyAssignmentsRequest();
        request.withPolicyAssignmentName("<policy_assignment_name>");
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ListPolicyAssignmentsResponse response = client.listPolicyAssignments(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \
```

```
client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListPolicyAssignmentsRequest()
    request.policy_assignment_name = "<policy_assignment_name>"
    request.limit = <limit>
    request.marker = "<marker>"
    response = client.list_policy_assignments(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPolicyAssignmentsRequest{
        policyAssignmentNameRequest:= "<policy_assignment_name>"
        request.PolicyAssignmentName = &policyAssignmentNameRequest
        limitRequest:= int32(<limit>)
        request.Limit = &limitRequest
        markerRequest:= "<marker>"
        request.Marker = &markerRequest
    }
    response, err := client.ListPolicyAssignments(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.5 Modifying a Rule

Function

This API is used to modify a rule.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}

Table 4-212 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-213 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-214 Request body parameters

Parameter	Mandatory	Type	Description
policy_assignment_type	No	String	Specifies the rule type, which can be builtin or custom .
name	Yes	String	Specifies the rule name. Maximum: 64
description	No	String	Specifies the rule description. Maximum: 512
period	No	String	Specifies the trigger period.
policy_filter	No	PolicyFilterDefinition object	Specifies the policy filter of a rule.
policy_definition_id	No	String	Specifies the policy definition ID. Maximum: 36
custom_policy	No	CustomPolicy object	Specifies the custom rule.
parameters	No	Map<String, PolicyParameterValue >	Specifies the rule parameters.
tags	No	Array of ResourceTag objects	

Table 4-215 PolicyFilterDefinition

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 128
resource_provider	No	String	Specifies the cloud service name. Maximum: 128
resource_type	No	String	Specifies the resource type. Maximum: 128
resource_id	No	String	Specifies the resource ID. Maximum: 512
tag_key	No	String	Specifies the tag key. Maximum: 128
tag_value	No	String	Specifies the tag value. Maximum: 256

Table 4-216 CustomPolicy

Parameter	Mandatory	Type	Description
function_urn	Yes	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	Yes	String	Specifies how function is called by a custom rule.
auth_value	No	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-217 PolicyParameterValue

Parameter	Mandatory	Type	Description
value	No	Object	Specifies the value of the rule parameter.

Table 4-218 ResourceTag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	Yes	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Response Parameters

Status code: 200

Table 4-219 Response body parameters

Parameter	Type	Description
policy_assignment_type	String	Specifies the rule type, which can be builtin or custom .
id	String	Specifies the rule ID.
name	String	Specifies the rule name.
description	String	Specifies the rule description.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
period	String	Specifies how often the rule is triggered, which can be One_Hour , Three_Hours , Six_Hours , Twelve_Hours , or TwentyFour_Hours .
state	String	Specifies the rule status.
created	String	Specifies the time when the rule was added.
updated	String	Specifies the time when the rule was modified.
policy_definition_id	String	Specifies the ID of the policy used by the rule.

Parameter	Type	Description
custom_policy	CustomPolicy object	Specifies the custom rule.
parameters	Map<String, PolicyParameter >	Specifies the rule parameters.
tags	Array of ResourceTag objects	
created_by	String	Specifies the person who created the rule.

Table 4-220 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-221 CustomPolicy

Parameter	Type	Description
function_urn	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	String	Specifies how function is called by a custom rule.
auth_value	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-222 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Table 4-223 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Status code: 400

Table 4-224 Response body parameters

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

Status code: 403

Table 4-225 Response body parameters

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

Status code: 404

Table 4-226 Response body parameters

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

Status code: 500

Table 4-227 Response body parameters

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

Example Requests

```
PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/policy-assignments/
{policy_assignment_id}

{
  "name" : "allowed-images-by-id",
  "description" : "The ECS resource is non-compliant if the image it used is not in the allowed list",
  "parameters" : {
    "listOfAllowedImages" : {
      "value" : [ "ea0d6e0e-99c3-406d-a873-3bb45462b624" ]
    }
  },
  "policy_filter" : {
    "resource_provider" : "ecs",
    "resource_type" : "cloudservers"
  },
  "policy_definition_id" : "5f3c8b9a9811a13739a32ed9"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "5fb76a240f2daf0b9662a04d",
  "name" : "volume-inuse-check",
  "description" : "An EVS disk is non-compliant if it is not mounted to a cloud server.",
  "policy_filter" : {
    "region_id" : null,
    "resource_provider" : "evs",
    "resource_type" : "volumes",
    "resource_id" : null,
    "tag_key" : null,
    "tag_value" : null
  },
  "state" : "Enabled",
  "created" : "2020-11-20T07:03:00",
  "updated" : "2020-11-20T07:03:00",
}
```

```
"policy_definition_id" : "5fa36558aa1e6afc05a3d0a7",  
"parameters" : { }  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
import java.util.Map;  
import java.util.HashMap;  
  
public class UpdatePolicyAssignmentSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new GlobalCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ConfigClient client = ConfigClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdatePolicyAssignmentRequest request = new UpdatePolicyAssignmentRequest();  
        PolicyAssignmentRequestBody body = new PolicyAssignmentRequestBody();  
        PolicyParameterValue parametersParameter = new PolicyParameterValue();  
        parametersParameter.withValue("[ea0d6e0e-99c3-406d-a873-3bb45462b624]");  
        Map<String, PolicyParameterValue> listbodyParameters = new HashMap<>();  
        listbodyParameters.put("listOfAllowedImages", parametersParameter);  
        PolicyFilterDefinition policyFilterbody = new PolicyFilterDefinition();  
        policyFilterbody.withResourceProvider("ecs")  
            .withResourceType("cloudservers");  
        body.withParameters(listbodyParameters);  
        body.withPolicyDefinitionId("5f3c8b9a9811a13739a32ed9");  
        body.withPolicyFilter(policyFilterbody);  
        body.withDescription("The ECS resource is non-compliant if the image it used is not in the allowed  
list");  
        body.withName("allowed-images-by-id");  
        request.withBody(body);  
        try {  
            UpdatePolicyAssignmentResponse response = client.updatePolicyAssignment(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdatePolicyAssignmentRequest()
        parametersParameters = PolicyParameterValue(
            value="[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
        )
        listParametersbody = {
            "listOfAllowedImages": parametersParameters
        }
        policyFilterbody = PolicyFilterDefinition(
            resource_provider="ecs",
            resource_type="cloudservers"
        )
        request.body = PolicyAssignmentRequestBody(
            parameters=listParametersbody,
            policy_definition_id="5f3c8b9a9811a13739a32ed9",
            policy_filter=policyFilterbody,
            description="The ECS resource is non-compliant if the image it used is not in the allowed list",
            name="allowed-images-by-id"
        )
        response = client.update_policy_assignment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePolicyAssignmentRequest{
        var valueParameters interface{} = "[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
        parametersParameters := model.PolicyParameterValue{
            Value: &valueParameters,
        }
        var listParametersbody = map[string](model.PolicyParameterValue){
            "listOfAllowedImages": parametersParameters,
        }
        resourceProviderPolicyFilter:= "ecs"
        resourceTypePolicyFilter:= "cloudservers"
        policyFilterbody := &model.PolicyFilterDefinition{
            ResourceProvider: &resourceProviderPolicyFilter,
            ResourceType: &resourceTypePolicyFilter,
        }
        policyDefinitionIdPolicyAssignmentRequestBody:= "5f3c8b9a9811a13739a32ed9"
        descriptionPolicyAssignmentRequestBody:= "The ECS resource is non-compliant if the image it used is
not in the allowed list"
        request.Body = &model.PolicyAssignmentRequestBody{
            Parameters: listParametersbody,
            PolicyDefinitionId: &policyDefinitionIdPolicyAssignmentRequestBody,
            PolicyFilter: policyFilterbody,
            Description: &descriptionPolicyAssignmentRequestBody,
            Name: "allowed-images-by-id",
        }
        response, err := client.UpdatePolicyAssignment(request)
        if err == nil {
            fmt.Printf("%+v\n", response)
        } else {
            fmt.Println(err)
        }
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.6 Querying a Rule

Function

This API is used to query a rule based on the rule ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/policy-assignments/
{policy_assignment_id}

Table 4-228 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-229 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-230 Response body parameters

Parameter	Type	Description
policy_assignment_type	String	Specifies the rule type, which can be builtin or custom .
id	String	Specifies the rule ID.
name	String	Specifies the rule name.
description	String	Specifies the rule description.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
period	String	Specifies how often the rule is triggered, which can be One_Hour , Three_Hours , Six_Hours , Twelve_Hours , or TwentyFour_Hours .
state	String	Specifies the rule status.
created	String	Specifies the time when the rule was added.
updated	String	Specifies the time when the rule was modified.
policy_definition_id	String	Specifies the ID of the policy used by the rule.
custom_policy	CustomPolicy object	Specifies the custom rule.
parameters	Map<String, PolicyParameter >	Specifies the rule parameters.

Parameter	Type	Description
tags	Array of ResourceTag objects	
created_by	String	Specifies the person who created the rule.

Table 4-231 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-232 CustomPolicy

Parameter	Type	Description
function_urn	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	String	Specifies how function is called by a custom rule.
auth_value	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-233 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Table 4-234 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Status code: 400

Table 4-235 Response body parameters

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

Status code: 403

Table 4-236 Response body parameters

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

Status code: 404

Table 4-237 Response body parameters

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

Status code: 500

Table 4-238 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id": "5fb76a240f2daf0b9662a04d",
  "name": "volume-inuse-check",
  "description": "An EVS disk is non-compliant if it is not mounted to a cloud server.",
  "policy_filter": {
    "region_id": null,
    "resource_provider": "evs",
    "resource_type": "volumes",
    "resource_id": null,
    "tag_key": null,
    "tag_value": null
  },
  "state": "Enabled",
  "created": "2020-11-20T07:03:00",
  "updated": "2020-11-20T07:03:00",
  "policy_definition_id": "5fa36558aa1e6afc05a3d0a7",
  "parameters": { }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowPolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
```

```

this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowPolicyAssignmentRequest request = new ShowPolicyAssignmentRequest();
try {
    ShowPolicyAssignmentResponse response = client.showPolicyAssignment(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowPolicyAssignmentRequest()
        response = client.show_policy_assignment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowPolicyAssignmentRequest{}
    response, err := client.ShowPolicyAssignment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.7 Deleting a Rule

Function

This API is used to delete a rule based on the rule ID.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}

Table 4-239 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-240 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-241 Response body parameters

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

Status code: 403

Table 4-242 Response body parameters

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

Status code: 404

Table 4-243 Response body parameters

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

Status code: 500

Table 4-244 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeletePolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        DeletePolicyAssignmentRequest request = new DeletePolicyAssignmentRequest();
        try {
            DeletePolicyAssignmentResponse response = client.deletePolicyAssignment(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = GlobalCredentials(ak, sk) \

client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = DeletePolicyAssignmentRequest()
    response = client.delete_policy_assignment(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeletePolicyAssignmentRequest{}
    response, err := client.DeletePolicyAssignment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.8 Enabling a Rule

Function

This API is used to enable a rule based on the rule ID.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/enable

Table 4-245 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-246 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-247 Response body parameters

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

Status code: 403

Table 4-248 Response body parameters

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

Status code: 404

Table 4-249 Response body parameters

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

Status code: 500

Table 4-250 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class EnablePolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        EnablePolicyAssignmentRequest request = new EnablePolicyAssignmentRequest();
        try {
            EnablePolicyAssignmentResponse response = client.enablePolicyAssignment(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        }
    }
}
```

```

    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = EnablePolicyAssignmentRequest()
        response = client.enable_policy_assignment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().

```

```

WithAk(ak).
WithSk(sk).
Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.EnablePolicyAssignmentRequest{}
response, err := client.EnablePolicyAssignment(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Operation accepted.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.9 Disabling a Rule

Function

This API is used to disable a rule based on the rule ID.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/disable

Table 4-251 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-252 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-253 Response body parameters

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

Status code: 403

Table 4-254 Response body parameters

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

Status code: 404

Table 4-255 Response body parameters

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

Status code: 500

Table 4-256 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DisablePolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```

```
ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
DisablePolicyAssignmentRequest request = new DisablePolicyAssignmentRequest();
try {
    DisablePolicyAssignmentResponse response = client.disablePolicyAssignment(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DisablePolicyAssignmentRequest()
        response = client.disable_policy_assignment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
```

```

config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DisablePolicyAssignmentRequest{}
    response, err := client.DisablePolicyAssignment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Operation accepted.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.10 Running a Resource Compliance Evaluation

Function

This API is used to trigger a rule manually by rule ID to evaluate resources.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/policy-states/run-evaluation

Table 4-257 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-258 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: **400**

Table 4-259 Response body parameters

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

Status code: 403

Table 4-260 Response body parameters

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

Status code: 404

Table 4-261 Response body parameters

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

Status code: 500

Table 4-262 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class RunEvaluationByPolicyAssignmentIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        RunEvaluationByPolicyAssignmentIdRequest request = new
        RunEvaluationByPolicyAssignmentIdRequest();
        try {
            RunEvaluationByPolicyAssignmentIdResponse response =
            client.runEvaluationByPolicyAssignmentId(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = GlobalCredentials(ak, sk) \

client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = RunEvaluationByPolicyAssignmentIdRequest()
    response = client.run_evaluation_by_policy_assignment_id(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RunEvaluationByPolicyAssignmentIdRequest{}
    response, err := client.RunEvaluationByPolicyAssignmentId(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Operation accepted.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.11 Querying the Evaluation Status of a Rule

Function

This API is used to query the evaluation status of a rule by rule ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/policy-states/evaluation-state

Table 4-263 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Request Parameters

Table 4-264 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-265 Response body parameters

Parameter	Type	Description
policy_assignment_id	String	Specifies the rule ID.
state	String	Specifies the evaluation status.
start_time	String	Specifies the evaluation start time.
end_time	String	Specifies the evaluation end time.
error_message	String	Specifies the error message for the evaluation failure.

Status code: 400

Table 4-266 Response body parameters

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

Status code: 403

Table 4-267 Response body parameters

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

Status code: 404

Table 4-268 Response body parameters

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

Status code: 500

Table 4-269 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "state": "Succeeded",
  "policy_assignment_id": "5fb6292765ee7f5e92a7ca4b",
  "start_time": "2020-11-19T08:13:27.441Z",
  "end_time": "2020-11-19T08:13:27.485Z",
  "error_message": null
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowEvaluationStateByAssignmentIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowEvaluationStateByAssignmentIdRequest request = new
        ShowEvaluationStateByAssignmentIdRequest();
        try {
            ShowEvaluationStateByAssignmentIdResponse response =
            client.showEvaluationStateByAssignmentId(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
```

```
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = GlobalCredentials(ak, sk) \

client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ShowEvaluationStateByAssignmentIdRequest()
    response = client.show_evaluation_state_by_assignment_id(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowEvaluationStateByAssignmentIdRequest{}
    response, err := client.ShowEvaluationStateByAssignmentId(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.12 Querying the Compliance Results of a Resource

Function

This API is used to query all compliance results of a resource evaluated by rules. The resource is searched by resource ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/policy-states

Table 4-270 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
resource_id	Yes	String	Specifies the resource ID. Maximum: 512

Table 4-271 Query Parameters

Parameter	Mandatory	Type	Description
compliance_state	No	String	Specifies the compliance status. Maximum: 16
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-272 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: **200**

Table 4-273 Response body parameters

Parameter	Type	Description
value	Array of PolicyState objects	Specifies the return value of querying the compliance result.
page_info	PageInfo object	Specifies the pagination object.

Table 4-274 PolicyState

Parameter	Type	Description
domain_id	String	Specifies the user ID.
region_id	String	Specifies the ID of the region the resource belongs to.
resource_id	String	Specifies the resource ID.
resource_name	String	Specifies the resource name.
resource_provider	String	Specifies the cloud service name.
resource_type	String	Specifies the resource type.
trigger_type	String	Specifies the trigger type. The value can be resource or period .
compliance_status	String	Specifies the compliance status.
policy_assignment_id	String	Specifies the rule ID.
policy_assignment_name	String	Specifies the rule name.
policy_definition_id	String	Specifies the policy ID.
evaluation_time	String	Specifies the evaluation time of compliance status.

Table 4-275 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-276 Response body parameters

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

Status code: 403

Table 4-277 Response body parameters

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

Status code: 404

Table 4-278 Response body parameters

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

Status code: 500

Table 4-279 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
```



```
"domain_id" : "059b5c937100d3e40ff0c00a7675a0a0",
"region_id" : "regionid1",
"resource_id" : "010d95bd-87cd-4f22-ac00-db7fba7d927e",
"resource_name" : "ecs-cc-image-test1",
"resource_provider" : "ecs",
"resource_type" : "cloudservers",
"compliance_state" : "NonCompliant",
"policy_assignment_id" : "5fb618a726a24c53767fa049",
"policy_assignment_name" : "policy-assignment-test1",
"policy_definition_id" : "5fa265c0aa1e6afc05a0ff07",
"evaluation_time" : 1605776482523
}],
"page_info" : {
  "current_count" : 1,
  "next_marker" : null
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListPolicyStatesByResourceIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPolicyStatesByResourceIdRequest request = new ListPolicyStatesByResourceIdRequest();
        request.withComplianceState("<compliance_state>");
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ListPolicyStatesByResourceIdResponse response = client.listPolicyStatesByResourceId(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListPolicyStatesByResourceIdRequest()
        request.compliance_state = "<compliance_state>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_policy_states_by_resource_id(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
```

```

WithSk(sk).
Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListPolicyStatesByResourceIdRequest{}
complianceStateRequest:= "<compliance_state>"
request.ComplianceState = &complianceStateRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
response, err := client.ListPolicyStatesByResourceId(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.13 Querying All Compliance Results of Resources Evaluated by a Rule

Function

This API is used to query all compliance results of resources evaluated by a rule. The rule is searched by rule ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/policy-states

Table 4-280 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Table 4-281 Query Parameters

Parameter	Mandatory	Type	Description
compliance_state	No	String	Specifies the compliance status. Maximum: 16
resource_id	No	String	Specifies the resource ID. Maximum: 512
resource_name	No	String	Specifies the resource name. Maximum: 256
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-282 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-283 Response body parameters

Parameter	Type	Description
value	Array of PolicyState objects	Specifies the return value of querying the compliance result.
page_info	PageInfo object	Specifies the pagination object.

Table 4-284 PolicyState

Parameter	Type	Description
domain_id	String	Specifies the user ID.
region_id	String	Specifies the ID of the region the resource belongs to.
resource_id	String	Specifies the resource ID.
resource_name	String	Specifies the resource name.
resource_provider	String	Specifies the cloud service name.
resource_type	String	Specifies the resource type.
trigger_type	String	Specifies the trigger type. The value can be resource or period .

Parameter	Type	Description
compliance_state	String	Specifies the compliance status.
policy_assignment_id	String	Specifies the rule ID.
policy_assignment_name	String	Specifies the rule name.
policy_definition_id	String	Specifies the policy ID.
evaluation_time	String	Specifies the evaluation time of compliance status.

Table 4-285 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-286 Response body parameters

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

Status code: 403

Table 4-287 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 404

Table 4-288 Response body parameters

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

Status code: 500

Table 4-289 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "domain_id": "059b5c937100d3e40ff0c00a7675a0a0",
    "region_id": "regionid1",
    "resource_id": "010d95bd-87cd-4f22-ac00-db7fba7d927e",
    "resource_name": "ecs-cc-image-test1",
    "resource_provider": "ecs",
    "resource_type": "cloudservers",
    "compliance_state": "NonCompliant",
    "policy_assignment_id": "5fb618a726a24c53767fa049",
    "policy_assignment_name": "policy-assignment-test1",
    "policy_definition_id": "5fa265c0aa1e6afc05a0ff07",
    "evaluation_time": 1605776482523
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListPolicyStatesByAssignmentIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPolicyStatesByAssignmentIdRequest request = new ListPolicyStatesByAssignmentIdRequest();
        request.withComplianceState("<compliance_state>");
        request.withResourceId("<resource_id>");
        request.withResourceName("<resource_name>");
        request.withLimit("<limit>");
        request.withMarker("<marker>");
        try {
            ListPolicyStatesByAssignmentIdResponse response = client.listPolicyStatesByAssignmentId(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *
```



```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListPolicyStatesByAssignmentIdRequest()
        request.compliance_state = "<compliance_state>"
        request.resource_id = "<resource_id>"
        request.resource_name = "<resource_name>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_policy_states_by_assignment_id(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPolicyStatesByAssignmentIdRequest{}
    complianceStateRequest := "<compliance_state>"
    request.ComplianceState = &complianceStateRequest
    resourceIdRequest := "<resource_id>"
    request.ResourceId = &resourceIdRequest
```

```

resourceNameRequest:= "<resource_name>"
request.ResourceName = &resourceNameRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
response, err := client.ListPolicyStatesByAssignmentId(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	No rules found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.14 Querying All Compliance Results of Resources Under a User

Function

This API is used to query all compliance results of resources under a user.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/policy-states

Table 4-290 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-291 Query Parameters

Parameter	Mandatory	Type	Description
compliance_state	No	String	Specifies the compliance status. Maximum: 16
resource_id	No	String	Specifies the resource ID. Maximum: 512
resource_name	No	String	Specifies the resource name. Maximum: 256
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-292 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-293 Response body parameters

Parameter	Type	Description
value	Array of PolicyState objects	Specifies the return value of querying the compliance result.
page_info	PageInfo object	Specifies the pagination object.

Table 4-294 PolicyState

Parameter	Type	Description
domain_id	String	Specifies the user ID.
region_id	String	Specifies the ID of the region the resource belongs to.
resource_id	String	Specifies the resource ID.
resource_name	String	Specifies the resource name.
resource_provider	String	Specifies the cloud service name.
resource_type	String	Specifies the resource type.
trigger_type	String	Specifies the trigger type. The value can be resource or period .
compliance_state	String	Specifies the compliance status.
policy_assignment_id	String	Specifies the rule ID.
policy_assignment_name	String	Specifies the rule name.
policy_definition_id	String	Specifies the policy ID.
evaluation_time	String	Specifies the evaluation time of compliance status.

Table 4-295 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-296 Response body parameters

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

Status code: 403

Table 4-297 Response body parameters

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

Status code: 404

Table 4-298 Response body parameters

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

Status code: 500

Table 4-299 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "domain_id": "059b5c937100d3e40ff0c00a7675a0a0",
    "region_id": "regionid1",
    "resource_id": "010d95bd-87cd-4f22-ac00-db7fba7d927e",
    "resource_name": "ecs-cc-image-test1",
    "resource_provider": "ecs",
    "resource_type": "cloudservers",
    "compliance_state": "NonCompliant",
    "policy_assignment_id": "5fb618a726a24c53767fa049",
    "policy_assignment_name": "policy-assignment-test1",
    "policy_definition_id": "5fa265c0aa1e6afc05a0ff07",
    "evaluation_time": 1605776482523
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListPolicyStatesByDomainIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
    }
}
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ListPolicyStatesByDomainIdRequest request = new ListPolicyStatesByDomainIdRequest();
request.withComplianceState("<compliance_state>");
request.withResourceId("<resource_id>");
request.withResourceName("<resource_name>");
request.withLimit(<limit>);
request.withMarker("<marker>");
try {
    ListPolicyStatesByDomainIdResponse response = client.listPolicyStatesByDomainId(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListPolicyStatesByDomainIdRequest()
        request.compliance_state = "<compliance_state>"
        request.resource_id = "<resource_id>"
        request.resource_name = "<resource_name>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_policy_states_by_domain_id(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPolicyStatesByDomainIdRequest{}
    complianceStateRequest := "<compliance_state>"
    request.ComplianceState = &complianceStateRequest
    resourceIdRequest := "<resource_id>"
    request.ResourceId = &resourceIdRequest
    resourceNameRequest := "<resource_name>"
    request.ResourceName = &resourceNameRequest
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    response, err := client.ListPolicyStatesByDomainId(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.15 Updating the Compliance Result

Function

This API is used to update the compliance result of a custom rule.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/policy-states

Table 4-300 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-301 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-302 Request body parameters

Parameter	Mandatory	Type	Description
policy_resource	Yes	PolicyResource object	Specifies the resource.
trigger_type	Yes	String	Specifies the trigger type.
compliance_status	Yes	String	Specifies the compliance status.
policy_assignment_id	Yes	String	Specifies the policy rule id.
policy_assignment_name	No	String	Specifies the policy rule name.
evaluation_time	Yes	String	Specifies when a rule is used to evaluate the resource compliance.
evaluation_hash	Yes	String	Specifies the evaluation verification code.

Table 4-303 PolicyResource

Parameter	Mandatory	Type	Description
resource_id	No	String	Specifies the resource ID.
resource_name	No	String	Specifies the resource name.
resource_provider	No	String	Specifies the cloud service name.
resource_type	No	String	Specifies the resource type.
region_id	No	String	Specifies the region ID.
domain_id	No	String	Specifies the ID of the user to which the resource belongs.

Response Parameters

Status code: 200

Table 4-304 Response body parameters

Parameter	Type	Description
domain_id	String	Specifies the user ID.
region_id	String	Specifies the ID of the region the resource belongs to.
resource_id	String	Specifies the resource ID.
resource_name	String	Specifies the resource name.
resource_provider	String	Specifies the cloud service name.
resource_type	String	Specifies the resource type.
trigger_type	String	Specifies the trigger type. The value can be resource or period .
compliance_state	String	Specifies the compliance status.
policy_assignment_id	String	Specifies the rule ID.
policy_assignment_name	String	Specifies the rule name.
policy_definition_id	String	Specifies the policy ID.
evaluation_time	String	Specifies the evaluation time of compliance status.

Status code: 400

Table 4-305 Response body parameters

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

Status code: 403

Table 4-306 Response body parameters

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

Status code: 500

Table 4-307 Response body parameters

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

Example Requests

The reported compliance result by functionGraph is **NonCompliant**.

PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/policy-states

```
{
  "policy_resource" : {
    "domain_id" : "d0123456789",
    "region_id" : "global",
    "resource_id" : "abc0123456789",
    "resource_name" : "test_user",
    "resource_provider" : "iam",
    "resource_type" : "users"
  },
  "trigger_type" : "resource",
  "compliance_state" : "NonCompliant",
  "policy_assignment_id" : "abc0123456789abc",
  "policy_assignment_name" : "custom_policy",
  "evaluation_time" : 1667374060248,
  "evaluation_hash" : "89342b8f338165651991afb8bd471396"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "domain_id" : "d0123456789",
  "region_id" : "global",
  "resource_id" : "abc0123456789",
  "resource_name" : "test_user",
  "resource_provider" : "iam",
  "resource_type" : "users",
  "trigger_type" : "resource",
  "compliance_state" : "NonCompliant",
  "policy_assignment_id" : "abc0123456789abc",
  "policy_assignment_name" : "custom_policy",
  "policy_definition_id" : null,
}
```

```
"evaluation_time" : 1667374060248
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

The reported compliance result by functionGraph is **NonCompliant**.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class UpdatePolicyStateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdatePolicyStateRequest request = new UpdatePolicyStateRequest();
        PolicyStateRequestBody body = new PolicyStateRequestBody();
        PolicyResource policyResourcebody = new PolicyResource();
        policyResourcebody.withResourceId("abc0123456789")
            .withResourceName("test_user")
            .withResourceProvider("iam")
            .withResourceType("users")
            .withRegionId("global")
            .withDomainId("d0123456789");
        body.withEvaluationHash("89342b8f338165651991afb8bd471396");
        body.withEvaluationTime("1667374060248");
        body.withPolicyAssignmentName("custom_policy");
        body.withPolicyAssignmentId("abc0123456789abc");

        body.withComplianceState(PolicyStateRequestBody.ComplianceStateEnum.fromValue("NonCompliant"));
        body.withTriggerType(PolicyStateRequestBody.TriggerTypeEnum.fromValue("resource"));
        body.withPolicyResource(policyResourcebody);
        request.withBody(body);
        try {
            UpdatePolicyStateResponse response = client.updatePolicyState(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

```
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

The reported compliance result by functionGraph is **NonCompliant**.

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdatePolicyStateRequest()
        policyResourcebody = PolicyResource(
            resource_id="abc0123456789",
            resource_name="test_user",
            resource_provider="iam",
            resource_type="users",
            region_id="global",
            domain_id="d0123456789"
        )
        request.body = PolicyStateRequestBody(
            evaluation_hash="89342b8f338165651991afb8bd471396",
            evaluation_time="1667374060248",
            policy_assignment_name="custom_policy",
            policy_assignment_id="abc0123456789abc",
            compliance_state="NonCompliant",
            trigger_type="resource",
            policy_resource=policyResourcebody
        )
        response = client.update_policy_state(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

The reported compliance result by functionGraph is **NonCompliant**.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdatePolicyStateRequest{
        resourceIdPolicyResource:= "abc0123456789"
        resourceNamePolicyResource:= "test_user"
        resourceProviderPolicyResource:= "iam"
        resourceTypePolicyResource:= "users"
        regionIdPolicyResource:= "global"
        domainIdPolicyResource:= "d0123456789"
        policyResourcebody := &model.PolicyResource{
            ResourceId: &resourceIdPolicyResource,
            ResourceName: &resourceNamePolicyResource,
            ResourceProvider: &resourceProviderPolicyResource,
            ResourceType: &resourceTypePolicyResource,
            RegionId: &regionIdPolicyResource,
            DomainId: &domainIdPolicyResource,
        }
        policyAssignmentNamePolicyStateRequestBody:= "custom_policy"
        request.Body = &model.PolicyStateRequestBody{
            EvaluationHash: "89342b8f338165651991afb8bd471396",
            EvaluationTime: "1667374060248",
            PolicyAssignmentName: &policyAssignmentNamePolicyStateRequestBody,
            PolicyAssignmentId: "abc0123456789abc",
            ComplianceState: model.GetPolicyStateRequestBodyComplianceStateEnum().NON_COMPLIANT,
            TriggerType: model.GetPolicyStateRequestBodyTriggerTypeEnum().RESOURCE,
            PolicyResource: policyResourcebody,
        }
    }
    response, err := client.UpdatePolicyState(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.16 Creating an Organization Rule

Function

This API is used to create an organization rule.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/organizations/{organization_id}/policy-assignments

Table 4-308 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Request Parameters

Table 4-309 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-310 Request body parameters

Parameter	Mandatory	Type	Description
excluded_accounts	No	Array of strings	Specifies the accounts to be excluded from the rule. Maximum: 32
organization_policy_assignment_name	Yes	String	Specifies the organization rule name. Maximum: 60
managed_policy_assignment_metadata	No	ManagedPolicyAssignmentMetadata object	Specifies the metadata of a managed rule.

Table 4-311 ManagedPolicyAssignmentMetadata

Parameter	Mandatory	Type	Description
description	No	String	Specifies the rule description. Maximum: 512
period	No	String	Specifies the trigger period.
parameters	No	Map<String, PolicyParameterValue >	Specifies the input parameter.
policy_filter	No	PolicyFilterDefinition object	Specifies the policy filter of a rule.
policy_definition_id	Yes	String	Specifies the identifier of a predefined policy. Maximum: 36

Table 4-312 PolicyParameterValue

Parameter	Mandatory	Type	Description
value	No	Object	Specifies the value of the rule parameter.

Table 4-313 PolicyFilterDefinition

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 128
resource_provider	No	String	Specifies the cloud service name. Maximum: 128
resource_type	No	String	Specifies the resource type. Maximum: 128
resource_id	No	String	Specifies the resource ID. Maximum: 512
tag_key	No	String	Specifies the tag key. Maximum: 128
tag_value	No	String	Specifies the tag value. Maximum: 256

Response Parameters

Status code: 200

Table 4-314 Response body parameters

Parameter	Type	Description
owner_id	String	Specifies the organization rule creator.
organization_id	String	Specifies the organization ID.
organization_policy_assignment_urn	String	Specifies the unique ID of the organization rule.
organization_policy_assignment_id	String	Specifies the organization rule ID.

Parameter	Type	Description
organization_policy_assignment_name	String	Specifies the organization rule name.
description	String	Specifies the description information.
period	String	Specifies the trigger period.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
parameters	Map<String, PolicyParameterValue >	Specifies the rule parameters.
policy_definition_id	String	Specifies the policy ID.
created_at	String	Specifies the creation time.
updated_at	String	Specifies the update time.

Table 4-315 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-316 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Status code: 400

Table 4-317 Response body parameters

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

Status code: 403

Table 4-318 Response body parameters

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

Status code: 500

Table 4-319 Response body parameters

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

Example Requests

Creating or modifying an organization rule.

```
PUT https://{endpoint}/v1/resource-manager/organizations/{organization_id}/policy-assignments
{
  "organization_policy_assignment_name": "allowed-images-by-id",
  "managed_policy_assignment_metadata": {
    "description": "The ECS resource is non-compliant if the image it used is not in the allowed list.",
    "parameters": {
      "listOfAllowedImages": {
        "value": [ "ea0d6e0e-99c3-406d-a873-3bb45462b624" ]
      }
    }
  }
}
```

```

    },
    "policy_filter" : {
      "resource_provider" : "ecs",
      "resource_type" : "cloudservers"
    },
    "policy_definition_id" : "5fa265c0aa1e6afc05a0ff07"
  }
}

```

Example Responses

Status code: 200

Operation succeeded.

```

{
  "owner_id" : "e74e043fab784a45ad88f5ef6a4bcffc",
  "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
  "organization_policy_assignment_urn" :
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:organizationPolicyAssignment:o-
  doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
  "organization_policy_assignment_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
  "organization_policy_assignment_name" : "allowed-images-by-id",
  "description" : "The ECS resource is non-compliant if the image it used is not in the allowed list.",
  "period" : null,
  "policy_filter" : {
    "region_id" : null,
    "resource_provider" : "evs",
    "resource_type" : "volumes",
    "resource_id" : null,
    "tag_key" : null,
    "tag_value" : null
  },
  "policy_definition_id" : "5fa265c0aa1e6afc05a0ff07",
  "parameters" : { },
  "created_at" : "2022-01-18T06:37:54.432Z",
  "updated_at" : "2022-01-18T06:37:54.432Z"
}

```

SDK Sample Code

The SDK sample code is as follows.

Java

Creating or modifying an organization rule.

```

package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.Map;
import java.util.HashMap;

public class CreateOrganizationPolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or

```

```
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
CreateOrganizationPolicyAssignmentRequest request = new
CreateOrganizationPolicyAssignmentRequest();
OrganizationPolicyAssignmentRequest body = new OrganizationPolicyAssignmentRequest();
PolicyFilterDefinition policyFilterManagedPolicyAssignmentMetadata = new PolicyFilterDefinition();
policyFilterManagedPolicyAssignmentMetadata.withResourceProvider("ecs")
    .withResourceType("cloudservers");
PolicyParameterValue parametersParameters = new PolicyParameterValue();
parametersParameters.withValue("[ea0d6e0e-99c3-406d-a873-3bb45462b624]");
Map<String, PolicyParameterValue> listManagedPolicyAssignmentMetadataParameters = new
HashMap<>();
listManagedPolicyAssignmentMetadataParameters.put("listOfAllowedImages", parametersParameters);
ManagedPolicyAssignmentMetadata managedPolicyAssignmentMetadatabody = new
ManagedPolicyAssignmentMetadata();
managedPolicyAssignmentMetadatabody.withDescription("The ECS resource is non-compliant if the
image it used is not in the allowed list.")
    .withParameters(listManagedPolicyAssignmentMetadataParameters)
    .withPolicyFilter(policyFilterManagedPolicyAssignmentMetadata)
    .withPolicyDefinitionId("5fa265c0aa1e6afc05a0ff07");
body.withManagedPolicyAssignmentMetadata(managedPolicyAssignmentMetadatabody);
body.withOrganizationPolicyAssignmentName("allowed-images-by-id");
request.withBody(body);
try {
    CreateOrganizationPolicyAssignmentResponse response =
client.createOrganizationPolicyAssignment(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Creating or modifying an organization rule.

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = GlobalCredentials(ak, sk) \

client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateOrganizationPolicyAssignmentRequest()
    policyFilterManagedPolicyAssignmentMetadata = PolicyFilterDefinition(
        resource_provider="ecs",
        resource_type="cloudservers"
    )
    parametersParameters = PolicyParameterValue(
        value="[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
    )
    listParametersManagedPolicyAssignmentMetadata = {
        "listOfAllowedImages": parametersParameters
    }
    managedPolicyAssignmentMetadatabody = ManagedPolicyAssignmentMetadata(
        description="The ECS resource is non-compliant if the image it used is not in the allowed list.",
        parameters=listParametersManagedPolicyAssignmentMetadata,
        policy_filter=policyFilterManagedPolicyAssignmentMetadata,
        policy_definition_id="5fa265c0aa1e6afc05a0ff07"
    )
    request.body = OrganizationPolicyAssignmentRequest(
        managed_policy_assignment_metadata=managedPolicyAssignmentMetadatabody,
        organization_policy_assignment_name="allowed-images-by-id"
    )
    response = client.create_organization_policy_assignment(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Creating or modifying an organization rule.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

```

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateOrganizationPolicyAssignmentRequest{}
resourceProviderPolicyFilter:= "ecs"
resourceTypePolicyFilter:= "cloudservers"
policyFilterManagedPolicyAssignmentMetadata := &model.PolicyFilterDefinition{
    ResourceProvider: &resourceProviderPolicyFilter,
    ResourceType: &resourceTypePolicyFilter,
}
var valueParameters interface{} = "[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
parametersParameters := model.PolicyParameterValue{
    Value: &valueParameters,
}
var listParametersManagedPolicyAssignmentMetadata = map[string](model.PolicyParameterValue){
    "listOfAllowedImages": parametersParameters,
}
descriptionManagedPolicyAssignmentMetadata:= "The ECS resource is non-compliant if the image it
used is not in the allowed list."
managedPolicyAssignmentMetadatabody := &model.ManagedPolicyAssignmentMetadata{
    Description: &descriptionManagedPolicyAssignmentMetadata,
    Parameters: listParametersManagedPolicyAssignmentMetadata,
    PolicyFilter: policyFilterManagedPolicyAssignmentMetadata,
    PolicyDefinitionId: "5fa265c0aa1e6afc05a0ff07",
}
request.Body = &model.OrganizationPolicyAssignmentRequest{
    ManagedPolicyAssignmentMetadata: managedPolicyAssignmentMetadatabody,
    OrganizationPolicyAssignmentName: "allowed-images-by-id",
}
response, err := client.CreateOrganizationPolicyAssignment(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.17 Querying Organization Rules

Function

This API is used to query organization rules.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/policy-assignments

Table 4-320 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Table 4-321 Query Parameters

Parameter	Mandatory	Type	Description
organization_policy_assignment_id	No	String	Specifies the organization assignment id Maximum: 512
organization_policy_assignment_name	No	String	Specifies the organization rule name. Maximum: 60
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-322 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-323 Response body parameters

Parameter	Type	Description
organization_policy_assignments	Array of OrganizationPolicyAssignmentResponse objects	Specifies organization rules.
page_info	PageInfo object	Specifies the pagination object.

Table 4-324 OrganizationPolicyAssignmentResponse

Parameter	Type	Description
owner_id	String	Specifies the organization rule creator.
organization_id	String	Specifies the organization ID.
organization_policy_assignment_urn	String	Specifies the unique ID of the organization rule.
organization_policy_assignment_id	String	Specifies the organization rule ID.
organization_policy_assignment_name	String	Specifies the organization rule name.

Parameter	Type	Description
description	String	Specifies the description information.
period	String	Specifies the trigger period.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
parameters	Map<String, PolicyParameterValue >	Specifies the rule parameters.
policy_definition_id	String	Specifies the policy ID.
created_at	String	Specifies the creation time.
updated_at	String	Specifies the update time.

Table 4-325 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-326 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Table 4-327 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-328** Response body parameters

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

Status code: 403**Table 4-329** Response body parameters

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

Status code: 500**Table 4-330** Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "organization_policy_assignments": [ {
    "owner_id": "e74e043fab784a45ad88f5ef6a4bcffc",
    "organization_id": "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
    "organization_policy_assignment_urn":
      "rms::e74e043fab784a45ad88f5ef6a4bcffc:organizationPolicyAssignment:o-
      doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
    "organization_policy_assignment_id": "d02b7fa9d5a74e638c1402d0868f71fd",
    "organization_policy_assignment_name": "allowed-images-by-id",
    "description": "The ECS resource is non-compliant if the image it used is not in the allowed list",
    "period": null,
    "policy_filter": {
      "region_id": null,
      "resource_provider": "evs",
      "resource_type": "volumes",
      "resource_id": null,
      "tag_key": null,
      "tag_value": null
    },
    "policy_definition_id": "5fa265c0aa1e6afc05a0ff07",
    "parameters": { },
    "created_at": "2022-01-18T06:37:54.432Z",
    "updated_at": "2022-01-18T06:37:54.432Z"
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": "CAESIGQwMml3ZmE5ZDVhNzRlNm4yZjE0MDJkMDg2OGY3MWZkGgTnpS5u"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListOrganizationPolicyAssignmentsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
```

```
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ListOrganizationPolicyAssignmentsRequest request = new ListOrganizationPolicyAssignmentsRequest();
    request.withOrganizationPolicyAssignmentId("<organization_policy_assignment_id>");
    request.withOrganizationPolicyAssignmentName("<organization_policy_assignment_name>");
    request.withLimit(<limit>);
    request.withMarker("<marker>");
    try {
        ListOrganizationPolicyAssignmentsResponse response =
client.listOrganizationPolicyAssignments(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListOrganizationPolicyAssignmentsRequest()
        request.organization_policy_assignment_id = "<organization_policy_assignment_id>"
        request.organization_policy_assignment_name = "<organization_policy_assignment_name>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_organization_policy_assignments(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListOrganizationPolicyAssignmentsRequest{}
    organizationPolicyAssignmentIdRequest := "<organization_policy_assignment_id>"
    request.OrganizationPolicyAssignmentId = &organizationPolicyAssignmentIdRequest
    organizationPolicyAssignmentNameRequest := "<organization_policy_assignment_name>"
    request.OrganizationPolicyAssignmentName = &organizationPolicyAssignmentNameRequest
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    response, err := client.ListOrganizationPolicyAssignments(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.

Status Code	Description
500	Server error.

Error Codes

See [Error Codes](#).

4.5.18 Querying a Specific Organization Rule

Function

This API is used to query a specific organization rule.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/policy-assignments/{organization_policy_assignment_id}

Table 4-331 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34
organization_policy_assignment_id	Yes	String	Specifies the organization rule ID. Maximum: 32

Request Parameters

Table 4-332 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-333 Response body parameters

Parameter	Type	Description
owner_id	String	Specifies the organization rule creator.
organization_id	String	Specifies the organization ID.
organization_policy_assignment_urn	String	Specifies the unique ID of the organization rule.
organization_policy_assignment_id	String	Specifies the organization rule ID.
organization_policy_assignment_name	String	Specifies the organization rule name.
excluded_accounts	Array of strings	Specifies the accounts excluded from the rule.
description	String	Specifies the description information.
period	String	Specifies the trigger period.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
parameters	Map<String, PolicyParameterValue >	Specifies the rule parameters.
policy_definition_id	String	Specifies the policy ID.
created_at	String	Specifies the creation time.
updated_at	String	Specifies the update time.

Table 4-334 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128

Parameter	Type	Description
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-335 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Status code: 400

Table 4-336 Response body parameters

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

Status code: 403

Table 4-337 Response body parameters

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

Status code: 500

Table 4-338 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "owner_id" : "e74e043fab784a45ad88f5ef6a4bcffc",
  "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
  "organization_policy_assignment_urn" :
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:organizationPolicyAssignment:o-
  doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
  "organization_policy_assignment_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
  "organization_policy_assignment_name" : "allowed-images-by-id",
  "description" : "The ECS resource is non-compliant if the image it used is not in the allowed list.",
  "period" : null,
  "policy_filter" : {
    "region_id" : null,
    "resource_provider" : "evs",
    "resource_type" : "volumes",
    "resource_id" : null,
    "tag_key" : null,
    "tag_value" : null
  },
  "policy_definition_id" : "5fa265c0aa1e6afc05a0ff07",
  "parameters" : { },
  "created_at" : "2022-01-18T06:37:54.432Z",
  "updated_at" : "2022-01-18T06:37:54.432Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowOrganizationPolicyAssignmentSolution {
```

```

public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");

    ICredential auth = new GlobalCredentials()
        .withAk(ak)
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ShowOrganizationPolicyAssignmentRequest request = new
    ShowOrganizationPolicyAssignmentRequest();
    try {
        ShowOrganizationPolicyAssignmentResponse response =
    client.showOrganizationPolicyAssignment(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrMsg());
    }
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowOrganizationPolicyAssignmentRequest()
        response = client.show_organization_policy_assignment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)

```

```
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowOrganizationPolicyAssignmentRequest{}
    response, err := client.ShowOrganizationPolicyAssignment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.19 Deleting an Organization Rule

Function

This API is used to delete an organization rule.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/organizations/{organization_id}/policy-assignments/{organization_policy_assignment_id}

Table 4-339 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34
organization_policy_assignment_id	Yes	String	Specifies the organization rule ID. Maximum: 32

Request Parameters

Table 4-340 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-341 Response body parameters

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

Status code: 403

Table 4-342 Response body parameters

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

Status code: 500

Table 4-343 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;
```

```
public class DeleteOrganizationPolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        DeleteOrganizationPolicyAssignmentRequest request = new
        DeleteOrganizationPolicyAssignmentRequest();
        try {
            DeleteOrganizationPolicyAssignmentResponse response =
            client.deleteOrganizationPolicyAssignment(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteOrganizationPolicyAssignmentRequest()
        response = client.delete_organization_policy_assignment(request)
```



```
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteOrganizationPolicyAssignmentRequest{}
    response, err := client.DeleteOrganizationPolicyAssignment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.

Status Code	Description
500	Server error.

Error Codes

See [Error Codes](#).

4.5.20 Updating an Organization Rule

Function

Updating an Organization Rule

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/organizations/{organization_id}/policy-assignments/{organization_policy_assignment_id}

Table 4-344 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34
organization_policy_assignment_id	Yes	String	Specifies the organization rule ID. Maximum: 32

Request Parameters

Table 4-345 Request body parameters

Parameter	Mandatory	Type	Description
excluded_accounts	No	Array of strings	Specifies the accounts to be excluded from the rule. Maximum: 32

Parameter	Mandatory	Type	Description
organization_policy_assignment_name	Yes	String	Specifies the organization rule name. Maximum: 60
managed_policy_assignment_metadata	No	ManagedPolicyAssignmentMetadata object	Specifies the metadata of a managed rule.

Table 4-346 ManagedPolicyAssignmentMetadata

Parameter	Mandatory	Type	Description
description	No	String	Specifies the rule description. Maximum: 512
period	No	String	Specifies the trigger period.
parameters	No	Map<String, PolicyParameterValue >	Specifies the input parameter.
policy_filter	No	PolicyFilterDefinition object	Specifies the policy filter of a rule.
policy_definition_id	Yes	String	Specifies the identifier of a predefined policy. Maximum: 36

Table 4-347 PolicyParameterValue

Parameter	Mandatory	Type	Description
value	No	Object	Specifies the value of the rule parameter.

Table 4-348 PolicyFilterDefinition

Parameter	Mandatory	Type	Description
region_id	No	String	Specifies the region ID. Maximum: 128

Parameter	Mandatory	Type	Description
resource_provider	No	String	Specifies the cloud service name. Maximum: 128
resource_type	No	String	Specifies the resource type. Maximum: 128
resource_id	No	String	Specifies the resource ID. Maximum: 512
tag_key	No	String	Specifies the tag key. Maximum: 128
tag_value	No	String	Specifies the tag value. Maximum: 256

Response Parameters

Status code: 200

Table 4-349 Response body parameters

Parameter	Type	Description
owner_id	String	Specifies the organization rule creator.
organization_id	String	Specifies the organization ID.
organization_policy_assignment_urn	String	Specifies the unique ID of the organization rule.
organization_policy_assignment_id	String	Specifies the organization rule ID.
organization_policy_assignment_name	String	Specifies the organization rule name.
description	String	Specifies the description information.
period	String	Specifies the trigger period.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.

Parameter	Type	Description
parameters	Map<String,PolicyParameterValue>	Specifies the rule parameters.
policy_definition_id	String	Specifies the policy ID.
created_at	String	Specifies the creation time.
updated_at	String	Specifies the update time.

Table 4-350 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-351 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Status code: 400

Table 4-352 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 403

Table 4-353 Response body parameters

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

Status code: 500

Table 4-354 Response body parameters

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

Example Requests

Updating an Organization Rule

```
PUT https://{endpoint}/v1/resource-manager/organizations/{organization_id}/policy-assignments/
{organization_policy_assignment_id}

{
  "organization_policy_assignment_name": "allowed-images-by-id",
  "managed_policy_assignment_metadata": {
    "description": "The ECS resource is non-compliant if the image it used is not in the allowed list.",
    "parameters": {
      "listOfAllowedImages": {
        "value": [ "ea0d6e0e-99c3-406d-a873-3bb45462b624" ]
      }
    },
    "policy_filter": {
      "resource_provider": "ecs",
      "resource_type": "cloudservers"
    },
    "policy_definition_id": "5fa265c0aa1e6afc05a0ff07"
  }
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "owner_id" : "e74e043fab784a45ad88f5ef6a4bcffc",
  "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
  "organization_policy_assignment_urn" :
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:organizationPolicyAssignment:o-
  doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
  "organization_policy_assignment_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
  "organization_policy_assignment_name" : "allowed-images-by-id",
  "description" : "The ECS resource is non-compliant if the image it used is not in the allowed list.",
  "period" : null,
  "policy_filter" : {
    "region_id" : null,
    "resource_provider" : "evs",
    "resource_type" : "volumes",
    "resource_id" : null,
    "tag_key" : null,
    "tag_value" : null
  },
  "policy_definition_id" : "5fa265c0aa1e6afc05a0ff07",
  "parameters" : { },
  "created_at" : "2022-01-18T06:37:54.432Z",
  "updated_at" : "2022-01-18T06:37:54.432Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Updating an Organization Rule

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.Map;
import java.util.HashMap;

public class UpdateOrganizationPolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdateOrganizationPolicyAssignmentRequest request = new
        UpdateOrganizationPolicyAssignmentRequest();
    }
}
```

```

OrganizationPolicyAssignmentRequest body = new OrganizationPolicyAssignmentRequest();
PolicyFilterDefinition policyFilterManagedPolicyAssignmentMetadata = new PolicyFilterDefinition();
policyFilterManagedPolicyAssignmentMetadata.withResourceProvider("ecs")
    .withResourceType("cloudservers");
PolicyParameterValue parametersParameters = new PolicyParameterValue();
parametersParameters.withValue("[ea0d6e0e-99c3-406d-a873-3bb45462b624]");
Map<String, PolicyParameterValue> listManagedPolicyAssignmentMetadataParameters = new
HashMap<>();
listManagedPolicyAssignmentMetadataParameters.put("listOfAllowedImages", parametersParameters);
ManagedPolicyAssignmentMetadata managedPolicyAssignmentMetadatabody = new
ManagedPolicyAssignmentMetadata();
managedPolicyAssignmentMetadatabody.withDescription("The ECS resource is non-compliant if the
image it used is not in the allowed list.")
    .withParameters(listManagedPolicyAssignmentMetadataParameters)
    .withPolicyFilter(policyFilterManagedPolicyAssignmentMetadata)
    .withPolicyDefinitionId("5fa265c0aa1e6afc05a0ff07");
body.withManagedPolicyAssignmentMetadata(managedPolicyAssignmentMetadatabody);
body.withOrganizationPolicyAssignmentName("allowed-images-by-id");
request.withBody(body);
try {
    UpdateOrganizationPolicyAssignmentResponse response =
client.updateOrganizationPolicyAssignment(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
}

```

Python

Updating an Organization Rule

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateOrganizationPolicyAssignmentRequest()
        policyFilterManagedPolicyAssignmentMetadata = PolicyFilterDefinition(
            resource_provider="ecs",
            resource_type="cloudservers"

```



```
)
parametersParameters = PolicyParameterValue(
    value="[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
)
listParametersManagedPolicyAssignmentMetadata = {
    "listOfAllowedImages": parametersParameters
}
managedPolicyAssignmentMetadatabody = ManagedPolicyAssignmentMetadata(
    description="The ECS resource is non-compliant if the image it used is not in the allowed list.",
    parameters=listParametersManagedPolicyAssignmentMetadata,
    policy_filter=policyFilterManagedPolicyAssignmentMetadata,
    policy_definition_id="5fa265c0aa1e6afc05a0ff07"
)
request.body = OrganizationPolicyAssignmentRequest(
    managed_policy_assignment_metadata=managedPolicyAssignmentMetadatabody,
    organization_policy_assignment_name="allowed-images-by-id"
)
response = client.update_organization_policy_assignment(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Updating an Organization Rule

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateOrganizationPolicyAssignmentRequest{
        resourceProviderPolicyFilter:= "ecs"
        resourceTypePolicyFilter:= "cloudservers"
        policyFilterManagedPolicyAssignmentMetadata := &model.PolicyFilterDefinition{
            ResourceProvider: &resourceProviderPolicyFilter,
            ResourceType: &resourceTypePolicyFilter,
        }
    }
    var valueParameters interface{} = "[ea0d6e0e-99c3-406d-a873-3bb45462b624]"
    parametersParameters := model.PolicyParameterValue{
        Value: &valueParameters,
    }
```

```

}
var listParametersManagedPolicyAssignmentMetadata = map[string](model.PolicyParameterValue){
    "listOfAllowedImages": parametersParameters,
}
descriptionManagedPolicyAssignmentMetadata:= "The ECS resource is non-compliant if the image it
used is not in the allowed list."
managedPolicyAssignmentMetadatabody := &model.ManagedPolicyAssignmentMetadata{
    Description: &descriptionManagedPolicyAssignmentMetadata,
    Parameters: listParametersManagedPolicyAssignmentMetadata,
    PolicyFilter: policyFilterManagedPolicyAssignmentMetadata,
    PolicyDefinitionId: "5fa265c0aa1e6afc05a0ff07",
}
request.Body = &model.OrganizationPolicyAssignmentRequest{
    ManagedPolicyAssignmentMetadata: managedPolicyAssignmentMetadatabody,
    OrganizationPolicyAssignmentName: "allowed-images-by-id",
}
response, err := client.UpdateOrganizationPolicyAssignment(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.21 Querying Statuses of Organization Rules

Function

This API is used to querying statuses of organization rules.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/policy-assignment-statuses

Table 4-355 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Table 4-356 Query Parameters

Parameter	Mandatory	Type	Description
organization_policy_assignment_id	No	String	Specifies the organization assignment id Maximum: 512
organization_policy_assignment_name	No	String	Specifies the organization rule name. Maximum: 60
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-357 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-358 Response body parameters

Parameter	Type	Description
value	Array of OrganizationPolicyAssignmentStatusResponse objects	Specifies the statuses of organization rules.
page_info	PageInfo object	Specifies the pagination object.

Table 4-359 OrganizationPolicyAssignmentStatusResponse

Parameter	Type	Description
organization_policy_assignment_id	String	Specifies the organization rule ID.
organization_policy_assignment_name	String	Specifies the organization rule name.
organization_policy_assignment_status	String	Specifies the status of the organization rule.
error_code	String	Specifies the error code returned when an organization rule fails to be created or updated.
error_message	String	Specifies the error message displayed when an organization rule fails to be created or updated.
created_at	String	Specifies the creation time.
updated_at	String	Specifies the update time.

Table 4-360 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-361 Response body parameters

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

Status code: 403

Table 4-362 Response body parameters

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

Status code: 500

Table 4-363 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value" : [ {
    "organization_policy_assignment_id" : "9642977cef864971b3b00ad56621a15f",
    "organization_policy_assignment_name" : "allowed-images-by-id",
    "organization_policy_assignment_status" : "CREATE_SUCCESSFUL",
    "error_code" : "null,",
    "error_message" : "null,",
    "created_at" : "2023-01-17T09:38:26.165Z",
    "updated_at" : "2023-01-17T09:38:26.165Z"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : "CAESIGQwMml3ZmE5ZDVhNzRlNm4YzE0MDJkMDg2OGY3MWZkGgTnpS5u"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowOrganizationPolicyAssignmentStatusesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowOrganizationPolicyAssignmentStatusesRequest request = new
        ShowOrganizationPolicyAssignmentStatusesRequest();
        request.withOrganizationPolicyAssignmentId("<organization_policy_assignment_id>");
        request.withOrganizationPolicyAssignmentName("<organization_policy_assignment_name>");
        request.withLimit("<limit>");
        request.withMarker("<marker>");
        try {
            ShowOrganizationPolicyAssignmentStatusesResponse response =
```

```
client.showOrganizationPolicyAssignmentStatuses(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowOrganizationPolicyAssignmentStatusesRequest()
        request.organization_policy_assignment_id = "<organization_policy_assignment_id>"
        request.organization_policy_assignment_name = "<organization_policy_assignment_name>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.show_organization_policy_assignment_statuses(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

```

risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowOrganizationPolicyAssignmentStatusesRequest{
    organizationPolicyAssignmentIdRequest:= "<organization_policy_assignment_id>"
    request.OrganizationPolicyAssignmentId = &organizationPolicyAssignmentIdRequest
    organizationPolicyAssignmentNameRequest:= "<organization_policy_assignment_name>"
    request.OrganizationPolicyAssignmentName = &organizationPolicyAssignmentNameRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    response, err := client.ShowOrganizationPolicyAssignmentStatuses(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.5.22 Querying Details About the Rule Statuses of Each Member Account in an Organization

Function

This API is used to querying details about the rule statuses of each member account in an organization.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/policy-assignment-detailed-status

Table 4-364 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Table 4-365 Query Parameters

Parameter	Mandatory	Type	Description
organization_policy_assignment_name	No	String	Specifies the organization rule name. Maximum: 60
organization_policy_assignment_id	No	String	Specifies the organization assignment id Maximum: 512
domain_id	No	String	Specifies the account ID. Maximum: 36
status	No	String	Specifies the rule status of the member account. The value is case sensitive. Maximum: 20
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-366 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-367 Response body parameters

Parameter	Type	Description
value	Array of OrganizationPolicyAssignmentDetailedStatusResponse objects	Specifies the status details about organization rules.
page_info	PageInfo object	Specifies the pagination object.

Table 4-368 OrganizationPolicyAssignmentDetailedStatusResponse

Parameter	Type	Description
domain_id	String	Specifies the account ID.
policy_assignment_id	String	Specifies the rule ID.

Parameter	Type	Description
policy_assignment_name	String	Specifies the rule name.
member_account_policy_assignment_status	String	Specifies the status of the rule in a member account in an organization.
error_code	String	Specifies the error code returned when a rule fails to be created or updated.
error_message	String	Specifies the error message returned when a rule fails to be created or updated.
created_at	String	Specifies the creation time.
updated_at	String	Specifies the update time.

Table 4-369 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-370 Response body parameters

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

Status code: 403

Table 4-371 Response body parameters

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

Status code: 500

Table 4-372 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "domain_id": "3fd23817bd85480e8dc4f520075de348",
    "policy_assignment_id": "63c79416e34d812e56d3e2fc",
    "policy_assignment_name": "allowed-images-by-id",
    "member_account_policy_assignment_status": "CREATE_SUCCESSFUL",
    "error_code": "null",
    "error_message": "null",
    "created_at": "2023-01-17T09:38:26.165Z",
    "updated_at": "2023-01-17T09:38:26.165Z"
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": "CAESIGQwMmI3ZmE5ZDVhNzRlNjM4YzE0MDJkMDg2OGY3MWZkGgTnpS5u"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
```

```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowOrganizationPolicyAssignmentDetailedStatusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowOrganizationPolicyAssignmentDetailedStatusRequest request = new
        ShowOrganizationPolicyAssignmentDetailedStatusRequest();
        request.withOrganizationPolicyAssignmentName("<organization_policy_assignment_name>");
        request.withOrganizationPolicyAssignmentId("<organization_policy_assignment_id>");

        request.withStatus(ShowOrganizationPolicyAssignmentDetailedStatusRequest.StatusEnum.fromValue("<stat
        us>"));
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ShowOrganizationPolicyAssignmentDetailedStatusResponse response =
            client.showOrganizationPolicyAssignmentDetailedStatus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")
```

```
credentials = GlobalCredentials(ak, sk) \

client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ShowOrganizationPolicyAssignmentDetailedStatusRequest()
    request.organization_policy_assignment_name = "<organization_policy_assignment_name>"
    request.organization_policy_assignment_id = "<organization_policy_assignment_id>"
    request.status = "<status>"
    request.limit = <limit>
    request.marker = "<marker>"
    response = client.show_organization_policy_assignment_detailed_status(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowOrganizationPolicyAssignmentDetailedStatusRequest{
        organizationPolicyAssignmentNameRequest:= "<organization_policy_assignment_name>"
        request.OrganizationPolicyAssignmentName = &organizationPolicyAssignmentNameRequest
        organizationPolicyAssignmentIdRequest:= "<organization_policy_assignment_id>"
        request.OrganizationPolicyAssignmentId = &organizationPolicyAssignmentIdRequest
        statusRequest:=
    }
    model.GetShowOrganizationPolicyAssignmentDetailedStatusRequestStatusEnum().<STATUS>
    request.Status = &statusRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    response, err := client.ShowOrganizationPolicyAssignmentDetailedStatus(request)
    if err == nil {
```

```

    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.6 Region Management

4.6.1 Querying Regions Visible to You

Function

This API is used to query regions visible to you.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/regions

Table 4-373 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-374 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Specifies the language of the information to return. The default name is zh-cn , which can be zh-cn and en-us . Default: zh-cn
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-375 Response body parameters

Parameter	Type	Description
value	Array of Region objects	Specifies the list of region information.

Table 4-376 Region

Parameter	Type	Description
region_id	String	Specifies the region ID.
display_name	String	Specifies the display name.

Status code: 400

Table 4-377 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 403

Table 4-378 Response body parameters

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

Status code: 404

Table 4-379 Response body parameters

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

Status code: 500

Table 4-380 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "region_id": "regionid1",
    "display_name": "region1"
  } ]
}
```

```
}, {  
  "region_id" : "regionid2",  
  "display_name" : "region2"  
}]  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
public class ListRegionsSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new GlobalCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ConfigClient client = ConfigClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListRegionsRequest request = new ListRegionsRequest();  
        try {  
            ListRegionsResponse response = client.listRegions(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import GlobalCredentials  
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListRegionsRequest()
        response = client.list_regions(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListRegionsRequest{}
    response, err := client.ListRegions(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	User not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.7 Advanced Queries

4.7.1 Running Advanced Queries

Function

This API is used to run advanced queries.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/run-query

Table 4-381 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-382 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-383 Request body parameters

Parameter	Mandatory	Type	Description
expression	Yes	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096

Response Parameters

Status code: 200

Table 4-384 Response body parameters

Parameter	Type	Description
query_info	QueryInfo object	Specifies the ResourceQL query field.
results	Array of objects	Specifies the ResourceQL query result.

Table 4-385 QueryInfo

Parameter	Type	Description
select_fields	Array of strings	Specifies the ResourceQL query field.

Status code: 400

Table 4-386 Response body parameters

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

Status code: 403

Table 4-387 Response body parameters

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

Status code: 429

Table 4-388 Response body parameters

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

Status code: 500

Table 4-389 Response body parameters

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

Example Requests

- Querying IDs of all VMs under your account

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/run-query
{
  "expression": "select id from resources where provider = 'ecs' and type = 'cloudservers'"
}
```
- Querying 100-GB Elastic Volume Service (EVS) disks under your account

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/run-query
```

```
{
  "expression" : "select * from resources where provider = 'evs' and type = 'volumes' and properties.size = 100"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "query_info" : {
    "select_fields" : [ "id" ]
  },
  "results" : [ {
    "id" : "91252cc9-bfd9-0709-0912-56b397e0ba3f"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Querying IDs of all VMs under your account

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class RunQuerySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        RunQueryRequest request = new RunQueryRequest();
        QueryRunRequestBody body = new QueryRunRequestBody();
        body.withExpression("select id from resources where provider = 'ecs' and type = 'cloudservers'");
        request.withBody(body);
        try {
            RunQueryResponse response = client.runQuery(request);
            System.out.println(response.toString());
        }
    }
}
```

```

    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
}

```

- **Querying 100-GB Elastic Volume Service (EVS) disks under your account**

```

package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class RunQuerySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        RunQueryRequest request = new RunQueryRequest();
        QueryRunRequestBody body = new QueryRunRequestBody();
        body.withExpression("select * from resources where provider = 'evs' and type = 'volumes' and
properties.size = 100");
        request.withBody(body);
        try {
            RunQueryResponse response = client.runQuery(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
}

```


Python

- Querying IDs of all VMs under your account

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RunQueryRequest()
        request.body = QueryRunRequestBody(
            expression="select id from resources where provider = 'ecs' and type = 'cloudservers'"
        )
        response = client.run_query(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- Querying 100-GB Elastic Volume Service (EVS) disks under your account

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RunQueryRequest()
        request.body = QueryRunRequestBody(
            expression="select * from resources where provider = 'evs' and type = 'volumes' and
```

```
properties.size = 100"
    )
    response = client.run_query(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- Querying IDs of all VMs under your account

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RunQueryRequest{}
    request.Body = &model.QueryRunRequestBody{
        Expression: "select id from resources where provider = 'ecs' and type = 'cloudservers'",
    }
    response, err := client.RunQuery(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- Querying 100-GB Elastic Volume Service (EVS) disks under your account

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```

ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.RunQueryRequest{}
request.Body = &model.QueryRunRequestBody{
    Expression: "select * from resources where provider = 'evs' and type = 'volumes' and
properties.size = 100",
}
response, err := client.RunQuery(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
429	Limit exceeded.
500	Server error.

Error Codes

See [Error Codes](#).

4.7.2 Creating an Advanced Query

Function

This API is used to create an advanced query.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/stored-queries

Table 4-390 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-391 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-392 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64

Parameter	Mandatory	Type	Description
description	No	String	Specifies the ResourceQL description. Minimum: 0 Maximum: 512
expression	Yes	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096

Response Parameters

Status code: 200

Table 4-393 Response body parameters

Parameter	Type	Description
id	String	ResourceQL ID. Maximum: 256
name	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64
description	String	Specifies the ResourceQL description. Minimum: 0 Maximum: 512
expression	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096
created	String	Specifies when ResourceQL was created. Minimum: 1 Maximum: 64
updated	String	Specifies when ResourceQL was updated. Minimum: 1 Maximum: 64

Status code: 400

Table 4-394 Response body parameters

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

Status code: 403

Table 4-395 Response body parameters

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

Status code: 500

Table 4-396 Response body parameters

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

Example Requests

Creating an advanced query to query stopped ECSs

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/stored-queries
```

```
{
  "name" : "stopped-ecs",
  "description" : "Querying stopped ECSs",
  "expression" : "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'SHUTOFF'"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "62b9126566d400721efceffa",
  "name" : "stopped-ecs",
  "description" : "Querying stopped ECSs",
  "expression" : "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'SHUTOFF'",
  "created" : "2022-06-27T02:13:57.107Z",
}
```

```
"updated" : "2022-06-27T02:13:57.107Z"  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Creating an advanced query to query stopped ECSs

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
public class CreateStoredQuerySolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new GlobalCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ConfigClient client = ConfigClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreateStoredQueryRequest request = new CreateStoredQueryRequest();  
        StoredQueryRequestBody body = new StoredQueryRequestBody();  
        body.withExpression("SELECT id, name FROM resources WHERE provider = 'ecs' AND type =  
'cloudservers' AND properties.status = 'SHUTOFF'");  
        body.withDescription("Querying stopped ECSs");  
        body.withName("stopped-ecs");  
        request.withBody(body);  
        try {  
            CreateStoredQueryResponse response = client.createStoredQuery(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

Creating an advanced query to query stopped ECSs

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateStoredQueryRequest()
        request.body = StoredQueryRequestBody(
            expression="SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers'
AND properties.status = 'SHUTOFF",
            description="Querying stopped ECSs",
            name="stopped-ecs"
        )
        response = client.create_stored_query(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Creating an advanced query to query stopped ECSs

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
```



```
WithSk(sk).
Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateStoredQueryRequest{}
descriptionStoredQueryRequestBody:= "Querying stopped ECSs"
request.Body = &model.StoredQueryRequestBody{
    Expression: "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'SHUTOFF'",
    Description: &descriptionStoredQueryRequestBody,
    Name: "stopped-ecs",
}
response, err := client.CreateStoredQuery(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.7.3 Listing Advanced Queries

Function

This API is used to list all advanced queries.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/stored-queries

Table 4-397 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-398 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
name	No	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64

Request Parameters

Table 4-399 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-400 Response body parameters

Parameter	Type	Description
value	Array of StoredQuery objects	Specifies the list of advanced queries.
page_info	PageInfo object	Specifies the pagination object.

Table 4-401 StoredQuery

Parameter	Type	Description
id	String	ResourceQL ID. Maximum: 256
name	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64
description	String	Specifies the ResourceQL description. Minimum: 0 Maximum: 512
expression	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096
created	String	Specifies when ResourceQL was created. Minimum: 1 Maximum: 64
updated	String	Specifies when ResourceQL was updated. Minimum: 1 Maximum: 64

Table 4-402 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-403 Response body parameters

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

Status code: 403

Table 4-404 Response body parameters

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

Status code: 500

Table 4-405 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value" : [ {
    "id" : "62b9126566d400721efceffa",
    "name" : "stopped-ecs",
    "description" : "Querying stopped ECSs",
    "expression" : "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'SHUTOFF'",
    "created" : "2022-06-27T02:13:57.107Z",
    "updated" : "2022-06-27T02:13:57.107Z"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListStoredQueriesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListStoredQueriesRequest request = new ListStoredQueriesRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        request.withName("<name>");
        try {
            ListStoredQueriesResponse response = client.listStoredQueries(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListStoredQueriesRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.name = "<name>"
        response = client.list_stored_queries(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```

ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListStoredQueriesRequest{}
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
nameRequest:= "<name>"
request.Name = &nameRequest
response, err := client.ListStoredQueries(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.7.4 Querying an Advanced Query

Function

This API is used to query an advanced query.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}

Table 4-406 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
query_id	Yes	String	Specifies the query ID. Maximum: 36

Request Parameters

Table 4-407 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-408 Response body parameters

Parameter	Type	Description
id	String	ResourceQL ID. Maximum: 256
name	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64

Parameter	Type	Description
description	String	Specifies the ResourceQL description. Minimum: 0 Maximum: 512
expression	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096
created	String	Specifies when ResourceQL was created. Minimum: 1 Maximum: 64
updated	String	Specifies when ResourceQL was updated. Minimum: 1 Maximum: 64

Status code: 400**Table 4-409** Response body parameters

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

Status code: 403**Table 4-410** Response body parameters

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

Status code: 404**Table 4-411** Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 500

Table 4-412 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "62b9126566d400721efceffa",
  "name" : "stopped-ecs",
  "description" : "Querying stopped ECSs",
  "expression" : "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'SHUTOFF'",
  "created" : "2022-06-27T02:13:57.107Z",
  "updated" : "2022-06-27T02:13:57.107Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowStoredQuerySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```

```
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowStoredQueryRequest request = new ShowStoredQueryRequest();
try {
    ShowStoredQueryResponse response = client.showStoredQuery(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowStoredQueryRequest()
        response = client.show_stored_query(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowStoredQueryRequest{}
    response, err := client.ShowStoredQuery(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	No resource found.
500	Server error.

Error Codes

See [Error Codes](#).

4.7.5 Updating an Advanced Query

Function

This API is used to update a custom query.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}

Table 4-413 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
query_id	Yes	String	Specifies the query ID. Maximum: 36

Request Parameters

Table 4-414 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-415 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64
description	No	String	Specifies the ResourceQL description. Minimum: 0 Maximum: 512
expression	Yes	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096

Response Parameters

Status code: 200

Table 4-416 Response body parameters

Parameter	Type	Description
id	String	ResourceQL ID. Maximum: 256
name	String	Specifies the ResourceQL name. Minimum: 1 Maximum: 64
description	String	Specifies the ResourceQL description. Minimum: 0 Maximum: 512
expression	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096
created	String	Specifies when ResourceQL was created. Minimum: 1 Maximum: 64

Parameter	Type	Description
updated	String	Specifies when ResourceQL was updated. Minimum: 1 Maximum: 64

Status code: 400

Table 4-417 Response body parameters

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

Status code: 403

Table 4-418 Response body parameters

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

Status code: 404

Table 4-419 Response body parameters

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

Status code: 500

Table 4-420 Response body parameters

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

Example Requests

Updating the advanced query to query running ECSs

```
PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}
{
  "name" : "running-ecs",
  "description" : "Querying running ECSs",
  "expression" : "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'ACTIVE'"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "62b9126566d400721efceffa",
  "name" : "running-ecs",
  "description" : "Querying running ECSs",
  "expression" : "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
properties.status = 'ACTIVE'",
  "created" : "2022-06-27T02:13:57.107Z",
  "updated" : "2022-07-09T09:12:00.007Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Updating the advanced query to query running ECSs

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class UpdateStoredQuerySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
```



```
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
UpdateStoredQueryRequest request = new UpdateStoredQueryRequest();
StoredQueryRequestBody body = new StoredQueryRequestBody();
body.withExpression("SELECT id, name FROM resources WHERE provider = 'ecs' AND type =
'cloudservers' AND properties.status = 'ACTIVE'");
body.withDescription("Querying running ECSs");
body.withName("running-ecs");
request.withBody(body);
try {
    UpdateStoredQueryResponse response = client.updateStoredQuery(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Updating the advanced query to query running ECSs

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateStoredQueryRequest()
        request.body = StoredQueryRequestBody(
            expression="SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers'
AND properties.status = 'ACTIVE'",
            description="Querying running ECSs",
            name="running-ecs"
        )
        response = client.update_stored_query(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Updating the advanced query to query running ECSs

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateStoredQueryRequest{
        descriptionStoredQueryRequestBody:= "Querying running ECSs"
        request.Body = &model.StoredQueryRequestBody{
            Expression: "SELECT id, name FROM resources WHERE provider = 'ecs' AND type = 'cloudservers' AND
            properties.status = 'ACTIVE'",
            Description: &descriptionStoredQueryRequestBody,
            Name: "running-ecs",
        }
    }
    response, err := client.UpdateStoredQuery(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.

Status Code	Description
403	User authentication failed.
404	No resource found.
500	Server error.

Error Codes

See [Error Codes](#).

4.7.6 Deleting an Advanced Query

Function

This API is used to delete an advanced query.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}

Table 4-421 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
query_id	Yes	String	Specifies the query ID. Maximum: 36

Request Parameters

Table 4-422 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-423 Response body parameters

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

Status code: 403

Table 4-424 Response body parameters

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

Status code: 404

Table 4-425 Response body parameters

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

Status code: 500

Table 4-426 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeleteStoredQuerySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteStoredQueryRequest request = new DeleteStoredQueryRequest();
        try {
            DeleteStoredQueryResponse response = client.deleteStoredQuery(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteStoredQueryRequest()
        response = client.delete_stored_query(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

```

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.DeleteStoredQueryRequest{}
response, err := client.DeleteStoredQuery(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	No resource found.
500	Server error.

Error Codes

See [Error Codes](#).

4.7.7 Querying Schemas

Function

This API is used to query schemas used by advanced queries.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/schemas

Table 4-427 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-428 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-429 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the invoker's token.
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-430 Response body parameters

Parameter	Type	Description
value	Array of ResourceSchemaResponse objects	schemas object.
page_info	PageInfo object	Specifies the pagination object.

Table 4-431 ResourceSchemaResponse

Parameter	Type	Description
type	String	Specifies the resource type.
schema	Object	Specifies the schema content.

Table 4-432 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-433 Response body parameters

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

Status code: 403

Table 4-434 Response body parameters

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

Status code: 500**Table 4-435** Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "type": "aad.instances",
    "schema": {
      "expireTime": "int",
      "ispSpec": "string",
      "specType": "int",
      "basicBandwidth": "int",
      "elasticBandwidth": "int",
      "serviceBandwidth": "int"
    }
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListSchemasSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListSchemasRequest request = new ListSchemasRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ListSchemasResponse response = client.listSchemas(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
```

```
.build()

try:
    request = ListSchemasRequest()
    request.limit = <limit>
    request.marker = "<marker>"
    response = client.list_schemas(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListSchemasRequest{}
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    response, err := client.ListSchemas(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.8 Resource Aggregators

4.8.1 Creating a Resource Aggregator

Function

This API is used to create a resource aggregator.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/aggregators

Table 4-436 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-437 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-438 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_name	Yes	String	Specifies the name of a resource aggregator. Minimum: 1 Maximum: 64
aggregator_type	Yes	String	Specifies the resource aggregator type, which can be ACCOUNT or ORGANIZATION .
account_aggregation_sources	No	AccountAggregationSource object	Specifies the source account of the aggregated data.

Table 4-439 AccountAggregationSource

Parameter	Mandatory	Type	Description
domain_ids	No	Array of strings	Specifies the account list. Minimum: 1 Maximum: 36 Array Length: 1 - 30

Response Parameters

Status code: 200

Table 4-440 Response body parameters

Parameter	Type	Description
aggregator_name	String	Specifies the name of a resource aggregator.
aggregator_id	String	Specifies the resource aggregator ID.
aggregator_url	String	Specifies the resource aggregator identifier.
aggregator_type	String	Specifies the resource aggregator type.
account_aggregation_sources	AccountAggregationSource object	Specifies the source account of the aggregated data.
updated_at	String	Specifies the time when the resource aggregator was updated.
created_at	String	Specifies the time when the resource aggregator was created.

Table 4-441 AccountAggregationSource

Parameter	Type	Description
domain_ids	Array of strings	Specifies the account list. Minimum: 1 Maximum: 36 Array Length: 1 - 30

Status code: 400

Table 4-442 Response body parameters

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

Status code: 403

Table 4-443 Response body parameters

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

Status code: 500

Table 4-444 Response body parameters

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

Example Requests

Creating a resource aggregator

PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators

```
{
  "aggregator_name": "example_aggregator",
  "aggregator_type": "ACCOUNT",
  "account_aggregation_sources": {
    "domain_ids": [ "3fd23817bd85480e8dc4f520075de348" ]
  }
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregator_name": "example_aggregator",
  "aggregator_id": "7ca3447830bd49c9b630a99680eaf90f",
  "aggregator_urn":
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:aggregationAggregator:7ca3447830bd49c9b630a99680eaf90f",
  "aggregator_type": "ACCOUNT",
  "account_aggregation_sources": {
    "domain_ids": [ "3fd23817bd85480e8dc4f520075de348" ]
  },
  "updated_at": "2020-01-18T02:49:12.465Z",
  "created_at": "2020-01-18T02:49:12.465Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Creating a resource aggregator

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateConfigurationAggregatorSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateConfigurationAggregatorRequest request = new CreateConfigurationAggregatorRequest();
        ConfigurationAggregatorRequest body = new ConfigurationAggregatorRequest();
        List<String> listAccountAggregationSourcesDomainIds = new ArrayList<>();
        listAccountAggregationSourcesDomainIds.add("3fd23817bd85480e8dc4f520075de348");
        AccountAggregationSource accountAggregationSourcesbody = new AccountAggregationSource();
        accountAggregationSourcesbody.withDomainIds(listAccountAggregationSourcesDomainIds);
        body.withAccountAggregationSources(accountAggregationSourcesbody);
        body.withAggregatorType("ACCOUNT");
        body.withAggregatorName("example_aggregator");
        request.withBody(body);
        try {
            CreateConfigurationAggregatorResponse response = client.createConfigurationAggregator(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Creating a resource aggregator

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateConfigurationAggregatorRequest()
        listDomainIdsAccountAggregationSources = [
            "3fd23817bd85480e8dc4f520075de348"
        ]
        accountAggregationSourcesbody = AccountAggregationSource(
            domain_ids=listDomainIdsAccountAggregationSources
        )
        request.body = ConfigurationAggregatorRequest(
            account_aggregation_sources=accountAggregationSourcesbody,
            aggregator_type="ACCOUNT",
            aggregator_name="example_aggregator"
        )
        response = client.create_configuration_aggregator(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Creating a resource aggregator

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
```

```

WithSk(sk).
Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateConfigurationAggregatorRequest{}
var listDomainIdsAccountAggregationSources = []string{
    "3fd23817bd85480e8dc4f520075de348",
}
accountAggregationSourcesbody := &model.AccountAggregationSource{
    DomainIds: &listDomainIdsAccountAggregationSources,
}
request.Body = &model.ConfigurationAggregatorRequest{
    AccountAggregationSources: accountAggregationSourcesbody,
    AggregatorType: "ACCOUNT",
    AggregatorName: "example_aggregator",
}
response, err := client.CreateConfigurationAggregator(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.2 Querying Resource Aggregators

Function

This API is used to query resource aggregators.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/aggregators

Table 4-445 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-446 Query Parameters

Parameter	Mandatory	Type	Description
aggregator_name	No	String	Specifies the name of a resource aggregator. Minimum: 1 Maximum: 64
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-447 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-448 Response body parameters

Parameter	Type	Description
configuration_aggregators	Array of ConfigurationAggregatorResp objects	Specifies resource aggregators.
page_info	PageInfo object	Specifies the pagination object.

Table 4-449 ConfigurationAggregatorResp

Parameter	Type	Description
aggregator_name	String	Specifies the name of a resource aggregator.
aggregator_id	String	Specifies the resource aggregator ID.
aggregator_url	String	Specifies the resource aggregator identifier.
aggregator_type	String	Specifies the resource aggregator type.
account_aggregation_sources	AccountAggregationSource object	Specifies the source account of the aggregated data.
updated_at	String	Specifies the time when the resource aggregator was updated.
created_at	String	Specifies the time when the resource aggregator was created.

Table 4-450 AccountAggregationSource

Parameter	Type	Description
domain_ids	Array of strings	Specifies the account list. Minimum: 1 Maximum: 36 Array Length: 1 - 30

Table 4-451 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-452 Response body parameters

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

Status code: 403

Table 4-453 Response body parameters

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

Status code: 500

Table 4-454 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "configuration_aggregators" : [ {
    "aggregator_name" : "example_aggregator",
    "aggregator_id" : "2f0cc4991bac4940ae20a3630818589e",
    "aggregator_urn" :
    "rms::e74e043fab784a45ad88f5ef6a4bcffc:aggregationAggregator:2f0cc4991bac4940ae20a3630818589e",
    "aggregator_type" : "ACCOUNT",
    "account_aggregation_sources" : {
      "domain_ids" : [ "3fd23817bd85480e8dc4f520075de348" ]
    },
    "updated_at" : "2020-01-18T02:49:12.465Z",
    "created_at" : "2020-01-18T02:49:12.465Z"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : "CAESIDdjYTM0NDc4MzBiZDQ5YzliNjMwYTk5NjgwZWVmOTBmGgQ-0aw3"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListConfigurationAggregatorsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListConfigurationAggregatorsRequest request = new ListConfigurationAggregatorsRequest();
        request.withAggregatorName("<aggregator_name>");
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
```

```

        ListConfigurationAggregatorsResponse response = client.listConfigurationAggregators(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListConfigurationAggregatorsRequest()
        request.aggregator_name = "<aggregator_name>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_configuration_aggregators(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment

```



```

variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListConfigurationAggregatorsRequest{}
aggregatorNameRequest := "<aggregator_name>"
request.AggregatorName = &aggregatorNameRequest
limitRequest := int32(<limit>)
request.Limit = &limitRequest
markerRequest := "<marker>"
request.Marker = &markerRequest
response, err := client.ListConfigurationAggregators(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.3 Querying a Specific Resource Aggregator

Function

This API is used to query a specific resource aggregator.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}

Table 4-455 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
aggregator_id	Yes	String	Specifies the resource aggregator ID. Maximum: 32

Request Parameters

Table 4-456 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-457 Response body parameters

Parameter	Type	Description
aggregator_name	String	Specifies the name of a resource aggregator.
aggregator_id	String	Specifies the resource aggregator ID.
aggregator_url	String	Specifies the resource aggregator identifier.
aggregator_type	String	Specifies the resource aggregator type.

Parameter	Type	Description
account_aggregation_sources	AccountAggregationSource object	Specifies the source account of the aggregated data.
updated_at	String	Specifies the time when the resource aggregator was updated.
created_at	String	Specifies the time when the resource aggregator was created.

Table 4-458 AccountAggregationSource

Parameter	Type	Description
domain_ids	Array of strings	Specifies the account list. Minimum: 1 Maximum: 36 Array Length: 1 - 30

Status code: 400

Table 4-459 Response body parameters

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

Status code: 403

Table 4-460 Response body parameters

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

Status code: 500

Table 4-461 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregator_name" : "example_aggregator",
  "aggregator_id" : "2f0cc4991bac4940ae20a3630818589e",
  "aggregator_urn" :
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:aggregationAggregator:2f0cc4991bac4940ae20a3630818589e",
  "aggregator_type" : "ACCOUNT",
  "account_aggregation_sources" : {
    "domain_ids" : [ "3fd23817bd85480e8dc4f520075de348" ]
  },
  "updated_at" : "2020-01-18T02:49:12.465Z",
  "created_at" : "2020-01-18T02:49:12.465Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowConfigurationAggregatorSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
```

```
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ShowConfigurationAggregatorRequest request = new ShowConfigurationAggregatorRequest();
    try {
        ShowConfigurationAggregatorResponse response = client.showConfigurationAggregator(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowConfigurationAggregatorRequest()
        response = client.show_configuration_aggregator(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
```

```

)
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowConfigurationAggregatorRequest{}
    response, err := client.ShowConfigurationAggregator(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.4 Querying the Status of a Specific Resource Aggregator Account

Function

This API is used to query the status of a specific resource aggregator account. The status can be **authorized** or **unauthorized**. If the status is **unauthorized**, the status information contains an error code or message.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}/aggregator-sources-status

Table 4-462 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
aggregator_id	Yes	String	Specifies the resource aggregator ID. Maximum: 32

Table 4-463 Query Parameters

Parameter	Mandatory	Type	Description
update_status	No	String	Specifies the status of the aggregated source account. Maximum: 16
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-464 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-465 Response body parameters

Parameter	Type	Description
aggregated_source_statuses	Array of AggregatedSourceStatus objects	Specifies the statuses of resource aggregators.
page_info	PageInfo object	Specifies the pagination object.

Table 4-466 AggregatedSourceStatus

Parameter	Type	Description
last_error_code	String	Specifies the error code returned when the last resource aggregation for the source account fails.
last_error_message	String	Specifies the error message returned when the last resource aggregation for the source account fails.
last_update_status	String	Specifies the latest status of the source account.
last_update_time	String	Specifies the last update time of the source account.
source_id	String	Specifies the source account ID or organization.
source_type	String	Specifies the source account type, which can be ACCOUNT or ORGANIZATION .

Parameter	Type	Description
source_name	String	Account name

Table 4-467 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-468** Response body parameters

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

Status code: 403**Table 4-469** Response body parameters

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

Status code: 500**Table 4-470** Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregated_source_statuses" : [ {
    "last_error_code" : null,
    "last_error_message" : null,
    "last_update_status" : "SUCCEEDED",
    "last_update_time" : "2020-01-18T10:49:12.442+08:00",
    "source_id" : "3fd23817bd85480e8dc4f520075de348",
    "source_name" : "demo",
    "source_type" : "ACCOUNT"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : "CAESIDdjYTM0NDc4MzBiZDQ5YzZliNjMwYTk5NjgwZWVmOTBmGgQ-0aw3"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowConfigurationAggregatorSourcesStatusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowConfigurationAggregatorSourcesStatusRequest request = new
ShowConfigurationAggregatorSourcesStatusRequest();

request.withUpdateStatus(ShowConfigurationAggregatorSourcesStatusRequest.UpdateStatusEnum.fromValu
e("<update_status>"));
request.withLimit(<limit>);
request.withMarker("<marker>");
try {
    ShowConfigurationAggregatorSourcesStatusResponse response =
client.showConfigurationAggregatorSourcesStatus(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowConfigurationAggregatorSourcesStatusRequest()
        request.update_status = "<update_status>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.show_configuration_aggregator_sources_status(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowConfigurationAggregatorSourcesStatusRequest{}
    updateStatusRequest:=
model.GetShowConfigurationAggregatorSourcesStatusRequestUpdateStatusEnum().<UPDATE_STATUS>
    request.UpdateStatus = &updateStatusRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    response, err := client.ShowConfigurationAggregatorSourcesStatus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.5 Updating a Resource Aggregator

Function

This API is used to update a resource aggregator.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}

Table 4-471 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
aggregator_id	Yes	String	Specifies the resource aggregator ID. Maximum: 32

Request Parameters

Table 4-472 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-473 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_name	Yes	String	Specifies the name of a resource aggregator. Minimum: 1 Maximum: 64
aggregator_type	Yes	String	Specifies the resource aggregator type, which can be ACCOUNT or ORGANIZATION .
account_aggregation_sources	No	AccountAggregationSource object	Specifies the source account of the aggregated data.

Table 4-474 AccountAggregationSource

Parameter	Mandatory	Type	Description
domain_ids	No	Array of strings	Specifies the account list. Minimum: 1 Maximum: 36 Array Length: 1 - 30

Response Parameters

Status code: 200

Table 4-475 Response body parameters

Parameter	Type	Description
aggregator_name	String	Specifies the name of a resource aggregator.
aggregator_id	String	Specifies the resource aggregator ID.
aggregator_url	String	Specifies the resource aggregator identifier.
aggregator_type	String	Specifies the resource aggregator type.
account_aggregation_sources	AccountAggregationSource object	Specifies the source account of the aggregated data.

Parameter	Type	Description
updated_at	String	Specifies the time when the resource aggregator was updated.
created_at	String	Specifies the time when the resource aggregator was created.

Table 4-476 AccountAggregationSource

Parameter	Type	Description
domain_ids	Array of strings	Specifies the account list. Minimum: 1 Maximum: 36 Array Length: 1 - 30

Status code: 400

Table 4-477 Response body parameters

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

Status code: 403

Table 4-478 Response body parameters

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

Status code: 500

Table 4-479 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Example Requests

Updating a resource aggregator

```
PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}
```

```
{
  "aggregator_name": "example_aggregator",
  "aggregator_type": "ACCOUNT",
  "account_aggregation_sources": {
    "domain_ids": [ "3fd23817bd85480e8dc4f520075de348" ]
  }
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregator_name": "example_aggregator",
  "aggregator_id": "7ca3447830bd49c9b630a99680eaf90f",
  "aggregator_urn":
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:aggregationAggregator:7ca3447830bd49c9b630a99680eaf90f",
  "aggregator_type": "ACCOUNT",
  "account_aggregation_sources": {
    "domain_ids": [ "3fd23817bd85480e8dc4f520075de348" ]
  },
  "updated_at": "2020-01-18T02:49:12.465Z",
  "created_at": "2020-01-18T02:49:12.465Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Updating a resource aggregator

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateConfigurationAggregatorSolution {
```



```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");

    ICredential auth = new GlobalCredentials()
        .withAk(ak)
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    UpdateConfigurationAggregatorRequest request = new UpdateConfigurationAggregatorRequest();
    ConfigurationAggregatorRequest body = new ConfigurationAggregatorRequest();
    List<String> listAccountAggregationSourcesDomainIds = new ArrayList<>();
    listAccountAggregationSourcesDomainIds.add("3fd23817bd85480e8dc4f520075de348");
    AccountAggregationSource accountAggregationSourcesbody = new AccountAggregationSource();
    accountAggregationSourcesbody.withDomainIds(listAccountAggregationSourcesDomainIds);
    body.withAccountAggregationSources(accountAggregationSourcesbody);
    body.withAggregatorType("ACCOUNT");
    body.withAggregatorName("example_aggregator");
    request.withBody(body);
    try {
        UpdateConfigurationAggregatorResponse response =
client.updateConfigurationAggregator(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

Updating a resource aggregator

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
```

```
.with_region(ConfigRegion.value_of("<YOUR REGION>")) \  
.build()  
  
try:  
    request = UpdateConfigurationAggregatorRequest()  
    listDomainIdsAccountAggregationSources = [  
        "3fd23817bd85480e8dc4f520075de348"  
    ]  
    accountAggregationSourcesbody = AccountAggregationSource(  
        domain_ids=listDomainIdsAccountAggregationSources  
    )  
    request.body = ConfigurationAggregatorRequest(  
        account_aggregation_sources=accountAggregationSourcesbody,  
        aggregator_type="ACCOUNT",  
        aggregator_name="example_aggregator"  
    )  
    response = client.update_configuration_aggregator(request)  
    print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

Updating a resource aggregator

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"  
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := global.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := config.NewConfigClient(  
        config.ConfigClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.UpdateConfigurationAggregatorRequest{}  
    var listDomainIdsAccountAggregationSources = []string{  
        "3fd23817bd85480e8dc4f520075de348",  
    }  
    accountAggregationSourcesbody := &model.AccountAggregationSource{  
        DomainIds: &listDomainIdsAccountAggregationSources,  
    }  
    request.Body = &model.ConfigurationAggregatorRequest{  
        AccountAggregationSources: accountAggregationSourcesbody,  
        AggregatorType: "ACCOUNT",  
        AggregatorName: "example_aggregator",  
    }
```

```

}
response, err := client.UpdateConfigurationAggregator(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.6 Deleting a Resource Aggregator

Function

This API is used to delete a resource aggregator.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}

Table 4-480 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Maximum: 32

Request Parameters

Table 4-481 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeleteConfigurationAggregatorSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
DeleteConfigurationAggregatorRequest request = new DeleteConfigurationAggregatorRequest();
try {
    DeleteConfigurationAggregatorResponse response = client.deleteConfigurationAggregator(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteConfigurationAggregatorRequest()
        response = client.delete_configuration_aggregator(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteConfigurationAggregatorRequest{}
    response, err := client.DeleteConfigurationAggregator(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	Operation succeeded.

Error Codes

See [Error Codes](#).

4.8.7 Authorizing a Resource Aggregator Account

Function

This API is used to authorize a resource aggregator account to collect data from source accounts.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization

Table 4-482 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-483 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-484 Request body parameters

Parameter	Mandatory	Type	Description
authorized_account_id	Yes	String	Specifies the ID of the resource aggregator account to be authorized. Minimum: 1 Maximum: 36

Response Parameters

Status code: 200

Table 4-485 Response body parameters

Parameter	Type	Description
aggregation_authorized_urn	String	Specifies the authorization identifier of the resource aggregator account.
authorized_account_id	String	Specifies the ID of the authorized resource aggregator account.
created_at	String	Specifies the time when the resource aggregator account was authorized.

Status code: 400

Table 4-486 Response body parameters

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

Status code: 403

Table 4-487 Response body parameters

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

Status code: 500

Table 4-488 Response body parameters

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

Example Requests

Authorizing a resource aggregator account

```
PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization
{
  "authorized_account_id" : "3fd23817bd85480e8dc4f520075de348"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregation_authorization_urn" :
  "rms::e74e043fab784a45ad88f5ef6a4bcffc:aggregationAuthorization:fb51a91e798a44beb831c9be365aba4db",
  "authorized_account_id" : "3fd23817bd85480e8dc4f520075de348",
  "created_at" : "2020-01-18T02:49:12.465Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Authorizing a resource aggregator account

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class CreateAggregationAuthorizationSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateAggregationAuthorizationRequest request = new CreateAggregationAuthorizationRequest();
        AggregationAuthorizationRequest body = new AggregationAuthorizationRequest();
        body.withAuthorizedAccountId("3fd23817bd85480e8dc4f520075de348");
        request.withBody(body);
    }
}
```

```
try {
    CreateAggregationAuthorizationResponse response =
client.createAggregationAuthorization(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Authorizing a resource aggregator account

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateAggregationAuthorizationRequest()
        request.body = AggregationAuthorizationRequest(
            authorized_account_id="3fd23817bd85480e8dc4f520075de348"
        )
        response = client.create_aggregation_authorization(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Authorizing a resource aggregator account

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
```

```

"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateAggregationAuthorizationRequest{}
    request.Body = &model.AggregationAuthorizationRequest{
        AuthorizedAccountId: "3fd23817bd85480e8dc4f520075de348",
    }
    response, err := client.CreateAggregationAuthorization(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.8 Querying Authorized Resource Aggregators Accounts

Function

This API is used to query authorized resource aggregator accounts.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization

Table 4-489 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-490 Query Parameters

Parameter	Mandatory	Type	Description
account_id	No	String	Specifies the ID of the authorized resource aggregator account. Maximum: 36
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-491 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-492 Response body parameters

Parameter	Type	Description
aggregation_authorized_accounts	Array of AggregationAuthorizationResp objects	Specifies authorized resource aggregator accounts.
page_info	PageInfo object	Specifies the pagination object.

Table 4-493 AggregationAuthorizationResp

Parameter	Type	Description
aggregation_authorized_account_urn	String	Specifies the authorization identifier of the resource aggregator account.
authorized_account_id	String	Specifies the ID of the authorized resource aggregator account.
created_at	String	Specifies the time when the resource aggregator account was authorized.

Table 4-494 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-495 Response body parameters

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

Status code: 403

Table 4-496 Response body parameters

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

Status code: 500

Table 4-497 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregation_authorizations" : [ {
    "aggregation_authorization_urn" :
"rms:e74e043fab784a45ad88f5ef6a4bcffc:aggregationAuthorization:fb51a91e798a44beb831c9be365aba4db",
    "authorized_account_id" : "3fd23817bd85480e8dc4f520075de348",
    "created_at" : "2022-12-15T07:50:01.575Z"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : "CAESIDdjYTM0NDc4MzBiZDQ5YzliNjMwYTk5NjgwZWVmOTBmGgQ-0aw3"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListAggregationAuthorizationsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        ListAggregationAuthorizationsRequest request = new ListAggregationAuthorizationsRequest();
        request.withAccountId("<account_id>");
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ListAggregationAuthorizationsResponse response = client.listAggregationAuthorizations(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

```
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAggregationAuthorizationsRequest()
        request.account_id = "<account_id>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_aggregation_authorizations(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
```



```

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListAggregationAuthorizationsRequest{}
accountIdRequest:= "<account_id>"
request.AccountId = &accountIdRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
response, err := client.ListAggregationAuthorizations(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.9 Deleting Authorization for a Resource Aggregator Account

Function

This API is used to delete authorization for a specific resource aggregator account.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization/{authorized_account_id}

Table 4-498 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
authorized_account_id	Yes	String	Specifies the ID of the authorized resource aggregator account. Maximum: 36

Request Parameters

Table 4-499 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeleteAggregationAuthorizationSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteAggregationAuthorizationRequest request = new DeleteAggregationAuthorizationRequest();
        try {
            DeleteAggregationAuthorizationResponse response =
client.deleteAggregationAuthorization(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
```

```
.with_credentials(credentials) \  
.with_region(ConfigRegion.value_of("<YOUR REGION>")) \  
.build()  
  
try:  
    request = DeleteAggregationAuthorizationRequest()  
    response = client.delete_aggregation_authorization(request)  
    print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"  
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := global.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := config.NewConfigClient(  
        config.ConfigClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.DeleteAggregationAuthorizationRequest{}  
    response, err := client.DeleteAggregationAuthorization(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	Operation succeeded.

Error Codes

See [Error Codes](#).

4.8.10 Querying All Pending Aggregation Requests

Function

This API is used to query all pending aggregation requests.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/aggregators/pending-aggregation-request

Table 4-500 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-501 Query Parameters

Parameter	Mandatory	Type	Description
account_id	No	String	Specifies the ID of the authorized resource aggregator account. Maximum: 36
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-502 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-503 Response body parameters

Parameter	Type	Description
pending_aggregation_requests	Array of PendingAggregationRequest objects	Specifies pending aggregation requests.
page_info	PageInfo object	Specifies the pagination object.

Table 4-504 PendingAggregationRequest

Parameter	Type	Description
requester_account_id	String	Specifies the ID of the account that requests aggregated data.

Table 4-505 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-506 Response body parameters

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

Status code: 403

Table 4-507 Response body parameters

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

Status code: 500

Table 4-508 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "pending_aggregation_requests" : [ {
    "requester_account_id" : "e74e043fab784a45ad88f5ef6a4bcffc"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : "CAESIDdjYTM0NDc4MzBiZDQ5YzliNjMwYTk5NjgwZWZmOTBmGgQ-0aw3"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListPendingAggregationRequestsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListPendingAggregationRequestsRequest request = new ListPendingAggregationRequestsRequest();
        request.withAccountId("<account_id>");
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ListPendingAggregationRequestsResponse response =
            client.listPendingAggregationRequests(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```



```
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListPendingAggregationRequestsRequest()
        request.account_id = "<account_id>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.list_pending_aggregation_requests(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
```

```

WithSk(sk).
Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListPendingAggregationRequestsRequest{}
accountIdRequest:= "<account_id>"
request.AccountId = &accountIdRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
response, err := client.ListPendingAggregationRequests(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.11 Deleting Pending Authorization Requests

Function

This API is used to delete pending authorization requests.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/aggregators/pending-aggregation-request/{requester_account_id}

Table 4-509 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
requester_account_id	Yes	String	Specifies the ID of the account that requests data aggregation. Maximum: 36

Request Parameters

Table 4-510 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeletePendingAggregationRequestSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        DeletePendingAggregationRequest request = new DeletePendingAggregationRequest();
        try {
            DeletePendingAggregationRequestResponse response =
client.deletePendingAggregationRequest(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
```

```
.with_credentials(credentials) \  
.with_region(ConfigRegion.value_of("<YOUR REGION>")) \  
.build()  
  
try:  
    request = DeletePendingAggregationRequestRequest()  
    response = client.delete_pending_aggregation_request(request)  
    print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"  
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := global.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := config.NewConfigClient(  
        config.ConfigClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.DeletePendingAggregationRequestRequest{}  
    response, err := client.DeletePendingAggregationRequest(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	Operation succeeded.

Error Codes

See [Error Codes](#).

4.8.12 Querying the Number of Resources of a Resource Aggregator Account

Function

This API is used to query the number of resources of a resource aggregator account. The number can be counted by **filter** and **GroupByKey**.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/aggregate-discovered-resource-counts

Table 4-511 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-512 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-513 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
filter	No	ResourceCountsFilters object	Specifies the filter for counting resources.
group_by_key	Yes	String	Specifies the key used to group resource counts. Its value can be RESOURCE_TYPE or ACCOUNT_ID . Minimum: 1 Maximum: 16

Table 4-514 ResourceCountsFilters

Parameter	Mandatory	Type	Description
account_id	No	String	Specifies the account ID. Minimum: 1 Maximum: 36
resource_type	No	String	Specifies the resource type. Minimum: 1 Maximum: 32
region_id	No	String	Specifies the region ID. Minimum: 1 Maximum: 32

Response Parameters

Status code: 200

Table 4-515 Response body parameters

Parameter	Type	Description
group_by_key	String	Specifies the key for grouping resource counts.

Parameter	Type	Description
grouped_resource_counts	Array of GroupedResourceCount objects	Specifies the list of group resources.
total_discovered_resources	Integer	Specifies the total number of resources in a filtered resource aggregator account.

Table 4-516 GroupedResourceCount

Parameter	Type	Description
group_name	String	Specifies the group name.
resource_count	Integer	Specifies the number of resources.

Status code: 400

Table 4-517 Response body parameters

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

Status code: 403

Table 4-518 Response body parameters

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

Status code: 500

Table 4-519 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Example Requests

Querying the number of resources of a resource aggregator account

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/aggregate-discovered-resource-counts
```

```
{
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f",
  "group_by_key" : "DOMAIN"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "group_by_key" : "DOMAIN",
  "grouped_resource_counts" : [ {
    "group_name" : "3fd23817bd85480e8dc4f520075de348",
    "resource_count" : 23
  } ],
  "total_discovered_resources" : 23
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying the number of resources of a resource aggregator account

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowAggregateDiscoveredResourceCountsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

```
ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowAggregateDiscoveredResourceCountsRequest request = new
ShowAggregateDiscoveredResourceCountsRequest();
AggregateDiscoveredResourceCountsRequest body = new
AggregateDiscoveredResourceCountsRequest();
body.withGroupByKey("DOMAIN");
body.withAggregatorId("7ca3447830bd49c9b630a99680eaf90f");
request.withBody(body);
try {
    ShowAggregateDiscoveredResourceCountsResponse response =
client.showAggregateDiscoveredResourceCounts(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Querying the number of resources of a resource aggregator account

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAggregateDiscoveredResourceCountsRequest()
        request.body = AggregateDiscoveredResourceCountsRequest(
            group_by_key="DOMAIN",
            aggregator_id="7ca3447830bd49c9b630a99680eaf90f"
        )
        response = client.show_aggregate_discovered_resource_counts(request)
        print(response)
```

```
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Querying the number of resources of a resource aggregator account

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAggregateDiscoveredResourceCountsRequest{}
    request.Body = &model.AggregateDiscoveredResourceCountsRequest{
        GroupByKey: "DOMAIN",
        AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
    }
    response, err := client.ShowAggregateDiscoveredResourceCounts(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.

Status Code	Description
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.13 Querying Resources of a Resource Aggregator Account

Function

This API is used to query resources of a resource aggregator account.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/aggregate-discovered-resources

Table 4-520 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-521 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-522 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-523 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
filter	No	ResourcesFilters object	Specifies the filter for counting resources.
provider	No	String	Specifies the cloud service type. Minimum: 1 Maximum: 20
resource_type	No	String	Specifies the resource type. Minimum: 1 Maximum: 32

Table 4-524 ResourcesFilters

Parameter	Mandatory	Type	Description
account_id	No	String	Specifies the account ID. Minimum: 1 Maximum: 36
region_id	No	String	Specifies the region ID. Minimum: 1 Maximum: 36
resource_id	No	String	Specifies the resource ID. Minimum: 1 Maximum: 512
resource_name	No	String	Specifies the resource name. Minimum: 1 Maximum: 256

Response Parameters

Status code: 200

Table 4-525 Response body parameters

Parameter	Type	Description
resource_identifiers	Array of ResourceIdentifier objects	Specifies the resource information list.
page_info	PageInfo object	Specifies the pagination object.

Table 4-526 ResourceIdentifier

Parameter	Type	Description
resource_id	String	Specifies the resource ID. Minimum: 1 Maximum: 512
resource_name	String	Specifies the resource name. Minimum: 1 Maximum: 256

Parameter	Type	Description
provider	String	Specifies the cloud service type. Minimum: 1 Maximum: 20
type	String	Specifies the resource type. Minimum: 1 Maximum: 32
source_account_id	String	Specifies the source account ID. Minimum: 1 Maximum: 36
region_id	String	Specifies the region to which the resource belongs. Minimum: 1 Maximum: 36

Table 4-527 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-528 Response body parameters

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

Status code: 403

Table 4-529 Response body parameters

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

Status code: 500

Table 4-530 Response body parameters

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

Example Requests

Querying resources of a resource aggregator account

POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/aggregate-discovered-resources

```
{
  "aggregator_id": "7ca3447830bd49c9b630a99680eaf90f",
  "provider": "vpc"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "resource_identifiers": [ {
    "resource_id": "1a429d95-3723-4bfa-b54a-19f041f66fe5",
    "resource_name": "vpc1",
    "provider": "vpc",
    "type": "vpcs",
    "source_account_id": "3fd23817bd85480e8dc4f520075de348",
    "region_id": "region1"
  }, {
    "resource_id": "3d23d144-82e8-4c7c-b0f7-06e4f413d0ee",
    "resource_name": "vpc2",
    "provider": "vpc",
    "type": "vpcs",
    "source_account_id": "3fd23817bd85480e8dc4f520075de348",
    "region_id": "region2"
  } ],
  "page_info": {
    "current_count": 2,
    "next_marker": "CAESIDdjYTM0NDc4MzBiZDQ5YzliNjMwYTk5NjgwZWVmOTBmGgQ-0aw3"
  }
}
```


SDK Sample Code

The SDK sample code is as follows.

Java

Querying resources of a resource aggregator account

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListAggregateDiscoveredResourcesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListAggregateDiscoveredResourcesRequest request = new ListAggregateDiscoveredResourcesRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        AggregateDiscoveredResourcesRequest body = new AggregateDiscoveredResourcesRequest();
        body.withProvider("vpc");
        body.withAggregatorId("7ca3447830bd49c9b630a99680eaf90f");
        request.withBody(body);
        try {
            ListAggregateDiscoveredResourcesResponse response =
client.listAggregateDiscoveredResources(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Querying resources of a resource aggregator account

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAggregateDiscoveredResourcesRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.body = AggregateDiscoveredResourcesRequest(
            provider="vpc",
            aggregator_id="7ca3447830bd49c9b630a99680eaf90f"
        )
        response = client.list_aggregate_discovered_resources(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Querying resources of a resource aggregator account

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
```

```

WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.ListAggregateDiscoveredResourcesRequest{}
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
providerAggregateDiscoveredResourcesRequest:= "vpc"
request.Body = &model.AggregateDiscoveredResourcesRequest{
    Provider: &providerAggregateDiscoveredResourcesRequest,
    AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
}
response, err := client.ListAggregateDiscoveredResources(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.14 Querying Details About a Specific Resource in a Source Account

Function

This API is used to query details about a specific resource in a source account.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-resource-config

Table 4-531 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-532 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-533 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
resource_identifier	Yes	ResourceIdentifier object	Specifies the list of resource details.

Table 4-534 ResourceIdentifier

Parameter	Mandatory	Type	Description
resource_id	Yes	String	Specifies the resource ID. Minimum: 1 Maximum: 512

Parameter	Mandatory	Type	Description
resource_name	No	String	Specifies the resource name. Minimum: 1 Maximum: 256
provider	Yes	String	Specifies the cloud service type. Minimum: 1 Maximum: 20
type	Yes	String	Specifies the resource type. Minimum: 1 Maximum: 32
source_account_id	Yes	String	Specifies the source account ID. Minimum: 1 Maximum: 36
region_id	Yes	String	Specifies the region to which the resource belongs. Minimum: 1 Maximum: 36

Response Parameters

Status code: 200

Table 4-535 Response body parameters

Parameter	Type	Description
resource_id	String	Specifies the resource ID.
aggregator_id	String	Specifies the resource aggregator ID.
aggregator_domain_id	String	Specifies the resource aggregator account.
domain_id	String	Specifies the ID of the account to which the aggregated resource belongs.
ep_id	String	Specifies the enterprise project ID.
provider	String	Specifies the cloud service name.
type	String	Specifies the resource type.
name	String	Specifies the resource name.

Parameter	Type	Description
region_id	String	Specifies the region ID.
project_id	String	Specifies the project ID in IaaS OpenStack.
created	String	Specifies the time when the resource was created.
updated	String	Specifies the time when the resource was updated.
tags	Map<String,String>	Specifies the resource tag.
properties	Map<String,Object>	Specifies properties of the resource.

Status code: 400

Table 4-536 Response body parameters

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

Status code: 403

Table 4-537 Response body parameters

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

Status code: 500

Table 4-538 Response body parameters

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

Example Requests

Querying details about a specific resource in the source account

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-resource-config

{
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f",
  "resource_identifier" : {
    "resource_id" : "1a429d95-3723-4bfa-b54a-19f041f66fe5",
    "provider" : "vpc",
    "type" : "vpcs",
    "source_account_id" : "3fd23817bd85480e8dc4f520075de348",
    "region_id" : "regionid1"
  }
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "resource_id" : "1a429d95-3723-4bfa-b54a-19f041f66fe5",
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f",
  "aggregator_domain_id" : "e74e043fab784a45ad88f5ef6a4bcffc",
  "domain_id" : "3fd23817bd85480e8dc4f520075de348\\\"",
  "ep_id" : 0,
  "provider" : "vpc",
  "type" : "vpcs",
  "name" : "vpc-xxx",
  "region_id" : "regionid1",
  "project_id" : "377cd852c57e4e949fc27c4882310364",
  "created" : "2022-10-24T07:03:40.000+00:00",
  "updated" : "2022-10-24T07:04:17.000+00:00",
  "properties" : {
    "cidr" : "192.168.0.0/16",
    "status" : "OK"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying details about a specific resource in the source account

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowAggregateResourceConfigSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

```
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowAggregateResourceConfigRequest request = new ShowAggregateResourceConfigRequest();
AggregateResourceConfigRequest body = new AggregateResourceConfigRequest();
ResourceIdentifier resourceIdentifierbody = new ResourceIdentifier();
resourceIdentifierbody.withResourceId("1a429d95-3723-4bfa-b54a-19f041f66fe5")
    .withProvider("vpc")
    .withType("vpcs")
    .withSourceAccountId("3fd23817bd85480e8dc4f520075de348")
    .withRegionId("regionid1");
body.withResourceIdentifier(resourceIdentifierbody);
body.withAggregatorId("7ca3447830bd49c9b630a99680eaf90f");
request.withBody(body);
try {
    ShowAggregateResourceConfigResponse response = client.showAggregateResourceConfig(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Querying details about a specific resource in the source account

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()
```



```
try:
    request = ShowAggregateResourceConfigRequest()
    resourceidentifierbody = ResourceIdentifier(
        resource_id="1a429d95-3723-4bfa-b54a-19f041f66fe5",
        provider="vpc",
        type="vpcs",
        source_account_id="3fd23817bd85480e8dc4f520075de348",
        region_id="regionid1"
    )
    request.body = AggregateResourceConfigRequest(
        resource_identifier=resourceidentifierbody,
        aggregator_id="7ca3447830bd49c9b630a99680eaf90f"
    )
    response = client.show_aggregate_resource_config(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Querying details about a specific resource in the source account

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAggregateResourceConfigRequest{
        resourceidentifierbody := &model.ResourceIdentifier{
            ResourceId: "1a429d95-3723-4bfa-b54a-19f041f66fe5",
            Provider: "vpc",
            Type: "vpcs",
            SourceAccountId: "3fd23817bd85480e8dc4f520075de348",
            RegionId: "regionid1",
        }
    }
    request.Body = &model.AggregateResourceConfigRequest{
        ResourceIdentifier: resourceidentifierbody,
        AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
    }
    response, err := client.ShowAggregateResourceConfig(request)
```

```

if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.15 Performing an Advanced Query on a Specific Aggregator

Function

This API is used to perform an advanced query on a specific aggregator.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}/run-query

Table 4-539 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Maximum: 32

Request Parameters

Table 4-540 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-541 Request body parameters

Parameter	Mandatory	Type	Description
expression	Yes	String	Specifies the ResourceQL expression. Minimum: 1 Maximum: 4096

Response Parameters

Status code: 200

Table 4-542 Response body parameters

Parameter	Type	Description
query_info	QueryInfo object	Specifies the ResourceQL query field.
results	Array of objects	Specifies the ResourceQL query result.

Table 4-543 QueryInfo

Parameter	Type	Description
select_fields	Array of strings	Specifies the ResourceQL query field.

Status code: 400

Table 4-544 Response body parameters

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

Status code: 403

Table 4-545 Response body parameters

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

Status code: 500

Table 4-546 Response body parameters

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

Example Requests

Querying all ECS IDs in the current organization

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}/run-query
{
  "expression" : "select id from aggregator_resources where provider = 'ecs' and type = 'cloudservers'"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "query_info" : {
    "select_fields" : [ "id" ]
  },
  "results" : [ {
    "id" : "91252cc9-bfd9-0709-0912-56b397e0ba3f"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying all ECS IDs in the current organization

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class RunAggregateResourceQuerySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        RunAggregateResourceQueryRequest request = new RunAggregateResourceQueryRequest();
        QueryRunRequestBody body = new QueryRunRequestBody();
        body.withExpression("select id from aggregator_resources where provider = 'ecs' and type =
'cloudservers'");
        request.withBody(body);
        try {
            RunAggregateResourceQueryResponse response = client.runAggregateResourceQuery(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

Querying all ECS IDs in the current organization

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RunAggregateResourceQueryRequest()
        request.body = QueryRunRequestBody(
            expression="select id from aggregator_resources where provider = 'ecs' and type = 'cloudservers'"
        )
        response = client.run_aggregate_resource_query(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Querying all ECS IDs in the current organization

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```

example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.RunAggregateResourceQueryRequest{}
request.Body = &model.QueryRunRequestBody{
    Expression: "select id from aggregator_resources where provider = 'ecs' and type = 'cloudservers'",
}
response, err := client.RunAggregateResourceQuery(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.16 Querying the Compliance Summary of One or More Source Accounts in an Aggregator

Function

This API is used to query the number of compliant and non-compliant rules of one or more accounts in the aggregator.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-states/compliance-summary

Table 4-547 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-548 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-549 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
account_id	No	String	Specifies the source account ID. Minimum: 1 Maximum: 36
group_by_key	No	String	Specifies the key DOMAIN used to group resource counts. Minimum: 1 Maximum: 16

Response Parameters

Status code: 200

Table 4-550 Response body parameters

Parameter	Type	Description
results	Array of AggregatePolicyComplianceSummaryResult objects	Specifies the compliance summary results.

Table 4-551 AggregatePolicyComplianceSummaryResult

Parameter	Type	Description
resource_details	PolicyComplianceSummaryUnit object	Specifies the compliance summary details.
assignment_details	PolicyComplianceSummaryUnit object	Specifies the compliance summary details.
group_name	String	Specifies the group name
group_account_name	String	Account name

Table 4-552 PolicyComplianceSummaryUnit

Parameter	Type	Description
compliant_count	Integer	Specifies the number of compliant resources.
non_compliant_count	Integer	Specifies the number of non-compliant resources.

Status code: 400

Table 4-553 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 403

Table 4-554 Response body parameters

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

Status code: 500

Table 4-555 Response body parameters

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

Example Requests

Querying the compliance summary of one or more source accounts in an aggregator

POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-states/compliance-summary

```
{
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f",
  "group_by_key" : "DOMAIN"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "results" : [ {
    "resource_details" : {
      "compliant_count" : null,
      "non_compliant_count" : 70
    },
    "assignment_details" : {
      "compliant_count" : 0,
      "non_compliant_count" : 1
    }
  } ],
}
```

```
"group_name" : "39f4a9434e0c4bf3806e5aff54adea39",
"group_account_name" : "demo1"
}, {
  "resource_details" : {
    "compliant_count" : null,
    "non_compliant_count" : 0
  },
  "assignment_details" : {
    "compliant_count" : 1,
    "non_compliant_count" : 0
  },
  "group_name" : "789d4ac10eda4d9b8e090f5c2b1e4459",
  "group_account_name" : "demo2"
}]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying the compliance summary of one or more source accounts in an aggregator

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowAggregatePolicyStateComplianceSummarySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        ShowAggregatePolicyStateComplianceSummaryRequest request = new
        ShowAggregatePolicyStateComplianceSummaryRequest();
        AggregatePolicyStatesRequest body = new AggregatePolicyStatesRequest();
        body.withGroupByKey("DOMAIN");
        body.withAggregatorId("7ca3447830bd49c9b630a99680eaf90f");
        request.withBody(body);
        try {
            ShowAggregatePolicyStateComplianceSummaryResponse response =
            client.showAggregatePolicyStateComplianceSummary(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Querying the compliance summary of one or more source accounts in an aggregator

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAggregatePolicyStateComplianceSummaryRequest()
        request.body = AggregatePolicyStatesRequest(
            group_by_key="DOMAIN",
            aggregator_id="7ca3447830bd49c9b630a99680eaf90f"
        )
        response = client.show_aggregate_policy_state_compliance_summary(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Querying the compliance summary of one or more source accounts in an aggregator

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
```

```

    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAggregatePolicyStateComplianceSummaryRequest{
        groupByKeyAggregatePolicyStatesRequest:= "DOMAIN"
        request.Body = &model.AggregatePolicyStatesRequest{
            GroupByKey: &groupByKeyAggregatePolicyStatesRequest,
            AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
        }
    }
    response, err := client.ShowAggregatePolicyStateComplianceSummary(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.17 Querying Aggregated Rules

Function

This API is used to query aggregated rules. For a rule, the numbers of both compliant and non-compliant resources are included.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-assignments/compliance

Table 4-556 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-557 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-558 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-559 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
filter	No	AggregatePolicyAssignmentsFilters object	Specifies the filter of aggregated rules.

Table 4-560 AggregatePolicyAssignmentsFilters

Parameter	Mandatory	Type	Description
account_id	No	String	Specifies the source account ID. Minimum: 1 Maximum: 36
compliance_state	No	String	Specifies the compliance result.
policy_assignment_name	No	String	Specifies the rule name. Maximum: 64

Response Parameters

Status code: 200

Table 4-561 Response body parameters

Parameter	Type	Description
aggregate_policy_assignments	Array of AggregatePolicyAssignments objects	Specifies the aggregated rules.
page_info	PageInfo object	Specifies the pagination object.

Table 4-562 AggregatePolicyAssignments

Parameter	Type	Description
policy_assignment_id	String	Specifies the rule ID
policy_assignment_name	String	Specifies the rule name.
compliance	Compliance object	Specifies the compliance of a rule.
account_id	String	Specifies the source account ID.
account_name	String	Account name

Table 4-563 Compliance

Parameter	Type	Description
compliance_state	String	Specifies the compliance result.
resource_details	PolicyComplianceSummaryUnit object	Specifies the compliance summary details.

Table 4-564 PolicyComplianceSummaryUnit

Parameter	Type	Description
compliant_count	Integer	Specifies the number of compliant resources.
non_compliant_count	Integer	Specifies the number of non-compliant resources.

Table 4-565 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-566 Response body parameters

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

Status code: 403

Table 4-567 Response body parameters

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

Status code: 500

Table 4-568 Response body parameters

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

Example Requests

Querying aggregated rules

```
POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-assignments/compliance
```

```
{
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "aggregate_policy_assignments" : [ {
    "policy_assignment_id" : "62bdbe5913c6836025b1e3af",
    "policy_assignment_name" : "volumes-encrypted-check",
    "compliance" : {
      "compliance_state" : "Compliant",
      "resource_details" : {
        "compliant_count" : 18,
        "non_compliant_count" : 0
      }
    },
    "account_id" : "789d4ac10eda4d9b8e090f5c2b1e4459",
    "account_name" : "demo1"
  }, {
    "policy_assignment_id" : "631ea13f57e0427679a1794d",
    "policy_assignment_name" : "cts-lts-enable",
    "compliance" : {
      "compliance_state" : "NonCompliant",
      "resource_details" : {
        "compliant_count" : 28,
        "non_compliant_count" : 2
      }
    },
    "account_id" : "83240c3f2ec1465aaa0124c87810ce65",
    "account_name" : "demo2"
  } ],
  "page_info" : {
    "current_count" : 2,
    "next_marker" : "CAESIDdjYTM0NDc4MzBiZDQ5YzliNjMwYTk5NjgwZWVmOTBmGgQ-0aw3"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying aggregated rules

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListAggregateComplianceByPolicyAssignmentSolution {
```

```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");

    ICredential auth = new GlobalCredentials()
        .withAk(ak)
        .withSk(sk);

    ConfigClient client = ConfigClient.newBuilder()
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ListAggregateComplianceByPolicyAssignmentRequest request = new
ListAggregateComplianceByPolicyAssignmentRequest();
    request.withLimit(<limit>);
    request.withMarker("<marker>");
    AggregatePolicyAssignmentsRequest body = new AggregatePolicyAssignmentsRequest();
    body.withAggregatorId("7ca3447830bd49c9b630a99680eaf90f");
    request.withBody(body);
    try {
        ListAggregateComplianceByPolicyAssignmentResponse response =
client.listAggregateComplianceByPolicyAssignment(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

Querying aggregated rules

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()
```

```
try:
    request = ListAggregateComplianceByPolicyAssignmentRequest()
    request.limit = <limit>
    request.marker = "<marker>"
    request.body = AggregatePolicyAssignmentsRequest(
        aggregator_id="7ca3447830bd49c9b630a99680eaf90f"
    )
    response = client.list_aggregate_compliance_by_policy_assignment(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Querying aggregated rules

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAggregateComplianceByPolicyAssignmentRequest{}
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    request.Body = &model.AggregatePolicyAssignmentsRequest{
        AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
    }
    response, err := client.ListAggregateComplianceByPolicyAssignment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.18 Querying Compliance Results of Aggregated Resources

Function

This API is used to query resources evaluated by a specified aggregated rule and their compliance statuses. A compliance result contains the evaluated resource and its compliance status.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-states/compliance-details

Table 4-569 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-570 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

Table 4-571 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-572 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
account_id	No	String	Specifies the source account ID. Minimum: 1 Maximum: 36
compliance_state	No	String	Specifies the compliance result.
policy_assignment_name	No	String	Specifies the rule name. Maximum: 64

Parameter	Mandatory	Type	Description
resource_name	No	String	Specifies the resource name. Minimum: 1 Maximum: 256
resource_id	No	String	Specifies the resource ID. Minimum: 1 Maximum: 512

Response Parameters

Status code: 200

Table 4-573 Response body parameters

Parameter	Type	Description
policy_states	Array of PolicyState objects	Specifies the return value of querying the compliance result.
page_info	PageInfo object	Specifies the pagination object.

Table 4-574 PolicyState

Parameter	Type	Description
domain_id	String	Specifies the user ID.
region_id	String	Specifies the ID of the region the resource belongs to.
resource_id	String	Specifies the resource ID.
resource_name	String	Specifies the resource name.
resource_provider	String	Specifies the cloud service name.
resource_type	String	Specifies the resource type.
trigger_type	String	Specifies the trigger type. The value can be resource or period .
compliance_state	String	Specifies the compliance status.

Parameter	Type	Description
policy_assignment_id	String	Specifies the rule ID.
policy_assignment_name	String	Specifies the rule name.
policy_definition_id	String	Specifies the policy ID.
evaluation_time	String	Specifies the evaluation time of compliance status.

Table 4-575 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-576** Response body parameters

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

Status code: 403**Table 4-577** Response body parameters

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

Status code: 500

Table 4-578 Response body parameters

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

Example Requests

Querying compliance results of aggregated resources

POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-states/compliance-details

```
{
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f",
  "account_id" : "39f4a9434e0c4bf3806e5aff54adea39",
  "policy_assignment_name" : "required-tag-check"
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "policy_states" : [ {
    "domain_id" : "39f4a9434e0c4bf3806e5aff54adea39",
    "region_id" : "region1",
    "resource_id" : "04d5a1a31c314ebba8c73daa66c6cd12",
    "resource_name" : "group1",
    "resource_provider" : "iam",
    "resource_type" : "groups",
    "trigger_type" : "resource",
    "compliance_state" : "NonCompliant",
    "policy_assignment_id" : "64104fff86952b3af614cabb",
    "policy_assignment_name" : "required-tag-check",
    "policy_definition_id" : "5fa9f89b6eed194ccb2c04db",
    "evaluation_time" : 1678790655196
  }, {
    "domain_id" : "39f4a9434e0c4bf3806e5aff54adea39",
    "region_id" : "global",
    "resource_id" : "06c113bd3edf40eabc552d29a775b9cf",
    "resource_name" : "group2",
    "resource_provider" : "iam",
    "resource_type" : "groups",
    "trigger_type" : "resource",
    "compliance_state" : "NonCompliant",
    "policy_assignment_id" : "64104fff86952b3af614cabb",
    "policy_assignment_name" : "required-tag-check",
    "policy_definition_id" : "5fa9f89b6eed194ccb2c04db",
    "evaluation_time" : 1678790655197
  } ],
  "page_info" : {
    "current_count" : 2,
    "next_marker" : "CAESIDdjYTM0NDc4MzBiZDQ5YzliNjMwYTk5NjgwZWVmOTBmGgQ-0aw3"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying compliance results of aggregated resources

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowAggregateComplianceDetailsByPolicyAssignmentSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        ShowAggregateComplianceDetailsByPolicyAssignmentRequest request = new
        ShowAggregateComplianceDetailsByPolicyAssignmentRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        AggregateComplianceDetailRequest body = new AggregateComplianceDetailRequest();
        body.withPolicyAssignmentName("required-tag-check");
        body.withAccountId("39f4a9434e0c4bf3806e5aff54adea39");
        body.withAggregatorId("7ca3447830bd49c9b630a99680eaf90f");
        request.withBody(body);
        try {
            ShowAggregateComplianceDetailsByPolicyAssignmentResponse response =
            client.showAggregateComplianceDetailsByPolicyAssignment(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Querying compliance results of aggregated resources

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAggregateComplianceDetailsByPolicyAssignmentRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.body = AggregateComplianceDetailRequest(
            policy_assignment_name="required-tag-check",
            account_id="39f4a9434e0c4bf3806e5aff54adea39",
            aggregator_id="7ca3447830bd49c9b630a99680eaf90f"
        )
        response = client.show_aggregate_compliance_details_by_policy_assignment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Querying compliance results of aggregated resources

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
```

```

WithAk(ak).
WithSk(sk).
Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowAggregateComplianceDetailsByPolicyAssignmentRequest{}
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
policyAssignmentNameAggregateComplianceDetailRequest:= "required-tag-check"
accountIdAggregateComplianceDetailRequest:= "39f4a9434e0c4bf3806e5aff54adea39"
request.Body = &model.AggregateComplianceDetailRequest{
    PolicyAssignmentName: &policyAssignmentNameAggregateComplianceDetailRequest,
    AccountId: &accountIdAggregateComplianceDetailRequest,
    AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
}
response, err := client.ShowAggregateComplianceDetailsByPolicyAssignment(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.
500	Server error.

Error Codes

See [Error Codes](#).

4.8.19 Querying Details About a Specified Aggregated Rule

Function

This API is used to query details about a specified aggregated rule.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-assignment/detail

Table 4-579 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-580 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-581 Request body parameters

Parameter	Mandatory	Type	Description
aggregator_id	Yes	String	Specifies the resource aggregator ID. Minimum: 1 Maximum: 256
account_id	Yes	String	Specifies the source account ID. Minimum: 1 Maximum: 36
policy_assignment_id	Yes	String	Specifies the rule ID. Maximum: 36

Response Parameters

Status code: 200

Table 4-582 Response body parameters

Parameter	Type	Description
policy_assignment_type	String	Specifies the rule type, which can be builtin or custom .
id	String	Specifies the rule ID.
name	String	Specifies the rule name.
description	String	Specifies the rule description.
policy_filter	PolicyFilterDefinition object	Specifies the policy filter of a rule.
period	String	Specifies how often the rule is triggered, which can be One_Hour , Three_Hours , Six_Hours , Twelve_Hours , or TwentyFour_Hours .
state	String	Specifies the rule status.
created	String	Specifies the time when the rule was added.
updated	String	Specifies the time when the rule was modified.
policy_definition_id	String	Specifies the ID of the policy used by the rule.
custom_policy	CustomPolicy object	Specifies the custom rule.
parameters	Map<String, PolicyParameterValue >	Specifies the rule parameters.
tags	Array of ResourceTag objects	
created_by	String	Specifies the person who created the rule.

Table 4-583 PolicyFilterDefinition

Parameter	Type	Description
region_id	String	Specifies the region ID. Maximum: 128

Parameter	Type	Description
resource_provider	String	Specifies the cloud service name. Maximum: 128
resource_type	String	Specifies the resource type. Maximum: 128
resource_id	String	Specifies the resource ID. Maximum: 512
tag_key	String	Specifies the tag key. Maximum: 128
tag_value	String	Specifies the tag value. Maximum: 256

Table 4-584 CustomPolicy

Parameter	Type	Description
function_urn	String	Specifies the URN of a custom function. Maximum: 1024
auth_type	String	Specifies how function is called by a custom rule.
auth_value	Map<String, Object>	Specifies the parameter value for invoking a function by a custom rule.

Table 4-585 PolicyParameterValue

Parameter	Type	Description
value	Object	Specifies the value of the rule parameter.

Table 4-586 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128

Parameter	Type	Description
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Status code: 400

Table 4-587 Response body parameters

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

Status code: 403

Table 4-588 Response body parameters

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

Status code: 500

Table 4-589 Response body parameters

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

Example Requests

Querying details about a specified aggregated rule

POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-assignment/detail

```
{
  "aggregator_id" : "7ca3447830bd49c9b630a99680eaf90f",
  "account_id" : "39f4a9434e0c4bf3806e5aff54adea39",
```



```
"policy_assignment_id" : "64104fff86952b3af614cabb"  
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{  
  "policy_assignment_type" : "builtin",  
  "id" : "64104fff86952b3af614cabb",  
  "name" : "required-tag-check",  
  "description" : "A resource is non-compliant if it does not have a specified tag.",  
  "policy_filter" : {  
    "region_id" : null,  
    "resource_provider" : null,  
    "resource_type" : null,  
    "resource_id" : null,  
    "tag_key" : null,  
    "tag_value" : null  
  },  
  "period" : null,  
  "state" : "Enabled",  
  "created" : "2023-04-03T06:40:12.846Z",  
  "updated" : "2023-04-03T06:40:12.846Z",  
  "policy_definition_id" : "5fa9f89b6eed194ccb2c04db",  
  "created_by" : "customer"  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Querying details about a specified aggregated rule

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
public class ShowAggregatePolicyAssignmentDetailSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new GlobalCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ConfigClient client = ConfigClient.newBuilder()
```

```
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
    ShowAggregatePolicyAssignmentDetailRequest request = new
ShowAggregatePolicyAssignmentDetailRequest();
    AggregatePolicyAssignmentDetailRequest body = new AggregatePolicyAssignmentDetailRequest();
    body.withPolicyAssignmentId("64104fff86952b3af614cabb");
    body.withAccountId("39f4a9434e0c4bf3806e5aff54adea39");
    body.withAggregatordId("7ca3447830bd49c9b630a99680eaf90f");
    request.withBody(body);
    try {
        ShowAggregatePolicyAssignmentDetailResponse response =
client.showAggregatePolicyAssignmentDetail(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

Querying details about a specified aggregated rule

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.valueOf("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAggregatePolicyAssignmentDetailRequest()
        request.body = AggregatePolicyAssignmentDetailRequest(
            policy_assignment_id="64104fff86952b3af614cabb",
            account_id="39f4a9434e0c4bf3806e5aff54adea39",
            aggregatord_id="7ca3447830bd49c9b630a99680eaf90f"
        )
        response = client.show_aggregate_policy_assignment_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Querying details about a specified aggregated rule

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAggregatePolicyAssignmentDetailRequest{}
    request.Body = &model.AggregatePolicyAssignmentDetailRequest{
        PolicyAssignmentId: "64104fff86952b3af614cabb",
        AccountId: "39f4a9434e0c4bf3806e5aff54adea39",
        AggregatorId: "7ca3447830bd49c9b630a99680eaf90f",
    }
    response, err := client.ShowAggregatePolicyAssignmentDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	Authentication failed or you do not have the operation permissions.

Status Code	Description
500	Server error.

Error Codes

See [Error Codes](#).

4.9 Conformance Packages

4.9.1 Querying Conformance Packages

Function

This API is used to query conformance packages.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/conformance-packs

Table 4-590 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-591 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
conformance_pack_name	No	String	Specifies the conformance package name. Minimum: 1 Maximum: 64

Request Parameters

Table 4-592 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-593 Response body parameters

Parameter	Type	Description
value	Array of Conformance Pack objects	Specifies conformance packages queried.
page_info	PageInfo object	Specifies the pagination object.

Table 4-594 ConformancePack

Parameter	Type	Description
id	String	Specifies the ID of a conformance package.
name	String	Specifies the conformance package name.

Parameter	Type	Description
stack_id	String	Specifies the unique ID of a resource stack. Minimum: 1 Maximum: 64
stack_name	String	Specifies the name of a resource stack. Minimum: 1 Maximum: 64
deployment_id	String	Specifies the deployment ID.
created_at	String	Specifies the time when a conformance package was created.
updated_at	String	Specifies the time when a conformance package was updated.
template_key	String	The name of a predefined conformance package template.
template_uri	String	The OBS address of a conformance package template
agency_name	String	The name of an agency.
status	String	Specifies the deployment status of a conformance package.
error_message	String	Specifies the error message when you failed to deploy or delete a conformance package.
vars_structure	Array of VarsStructure objects	Specifies the parameters of a conformance package.
created_by	String	Creator

Table 4-595 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Table 4-596 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-597 Response body parameters

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

Status code: 403

Table 4-598 Response body parameters

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

Status code: 500

Table 4-599 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "id": "a0123456789",
    "name": "conformance_pack_name-e35r",
    "stack_id": "stack-id-1234",
    "stack_name": "stack-name-1234",
    "deployment_id": "deployment_id_1234",
    "created_at": "2023-02-28T09:39:19Z",
    "updated_at": "2023-02-28T09:39:19Z",
    "status": "CREATE_FAILED",
    "error_message": "policy assignments exceed quota",
    "vars_structure": [ {
      "var_key": "test",
      "var_value": "test"
    } ]
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListConformancePacksSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListConformancePacksRequest request = new ListConformancePacksRequest();
        request.withLimit(<limit>);
    }
}
```



```
request.withMarker("<marker>");
request.withConformancePacName("<conformance_pack_name>");
try {
    ListConformancePacksResponse response = client.listConformancePacks(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListConformancePacksRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.conformance_pack_name = "<conformance_pack_name>"
        response = client.list_conformance_packs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListConformancePacksRequest{}
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    conformancePackNameRequest := "<conformance_pack_name>"
    request.ConformancePackName = &conformancePackNameRequest
    response, err := client.ListConformancePacks(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.2 Creating a Conformance Package

Function

This API is used to create a conformance package.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/domains/{domain_id}/conformance-packs

Table 4-600 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Request Parameters

Table 4-601 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Table 4-602 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the conformance package name. Minimum: 0 Maximum: 64

Parameter	Mandatory	Type	Description
agency_name	No	String	Specifies the agency name. The agency needs to authorize RFS to invoke the APIs for creating, updating, deleting a rule. Minimum: 0 Maximum: 64
template_key	No	String	Specifies the name of a built-in conformance package template. Maximum: 128
template_body	No	String	Specifies the content of a custom conformance package. Minimum: 1 Maximum: 51200
template_uri	No	String	Specifies the URL address of the OBS bucket where a conformance package template was stored. Minimum: 0 Maximum: 1024
vars_structure	No	Array of VarsStructure objects	Specifies the parameters of a conformance package. Array Length: 0 - 50

Table 4-603 VarsStructure

Parameter	Mandatory	Type	Description
var_key	No	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	No	Object	Specifies the value of a parameter.

Response Parameters

Status code: **200**

Table 4-604 Response body parameters

Parameter	Type	Description
id	String	Specifies the ID of a conformance package.
name	String	Specifies the conformance package name.
stack_id	String	Specifies the unique ID of a resource stack. Minimum: 1 Maximum: 64
stack_name	String	Specifies the name of a resource stack. Minimum: 1 Maximum: 64
deployment_id	String	Specifies the deployment ID.
created_at	String	Specifies the time when a conformance package was created.
updated_at	String	Specifies the time when a conformance package was updated.
template_key	String	The name of a predefined conformance package template.
template_uri	String	The OBS address of a conformance package template
agency_name	String	The name of an agency.
status	String	Specifies the deployment status of a conformance package.
error_message	String	Specifies the error message when you failed to deploy or delete a conformance package.
vars_structure	Array of VarsStructure objects	Specifies the parameters of a conformance package.
created_by	String	Creator

Table 4-605 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Status code: 400

Table 4-606 Response body parameters

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

Status code: 403

Table 4-607 Response body parameters

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

Status code: 500

Table 4-608 Response body parameters

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

Example Requests

Create a conformance package based on a template.

POST https://{endpoint}/v1/resource-manager/domains/{domain_id}/conformance-packs

```
{
  "name" : "conformance_pack_name",
  "agency_name" : "config_conformance_pack_template",
  "template_key" : "example-001.yaml",
  "template_body" : null,
  "template_uri" : null,
  "vars_structure" : [ {
    "var_key" : "test",
    "var_value" : "test"
  } ]
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "a0123456789",
  "name" : "conformance_pack_name-e35r",
  "stack_id" : "stack-id-1234",
  "stack_name" : "stack-name-1234",
  "deployment_id" : "deployment_id_1234",
  "created_at" : "2023-02-28T09:39:19Z",
  "updated_at" : "2023-02-28T09:39:19Z",
  "status" : "CREATE_FAILED",
  "error_message" : "policy assignments exceed quota",
  "vars_structure" : [ {
    "var_key" : "test",
    "var_value" : "test"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Create a conformance package based on a template.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateConformancePackRequest request = new CreateConformancePackRequest();
        ConformancePackRequestBody body = new ConformancePackRequestBody();
        List<VarsStructure> listbodyVarsStructure = new ArrayList<>();
        listbodyVarsStructure.add(
            new VarsStructure()
                .withVarKey("test")
                .withVarValue("test")
        );
        body.withVarsStructure(listbodyVarsStructure);
        body.withTemplateKey("example-001.yaml");
    }
}
```

```
body.withAgencyName("config_conformance_pack_template");
body.withName("conformance_pack_name");
request.withBody(body);
try {
    CreateConformancePackResponse response = client.createConformancePack(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Create a conformance package based on a template.

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateConformancePackRequest()
        listVarsStructurebody = [
            VarsStructure(
                var_key="test",
                var_value="test"
            )
        ]
        request.body = ConformancePackRequestBody(
            vars_structure=listVarsStructurebody,
            template_key="example-001.yaml",
            agency_name="config_conformance_pack_template",
            name="conformance_pack_name"
        )
        response = client.create_conformance_pack(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

Create a conformance package based on a template.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateConformancePackRequest{}
    varKeyVarsStructure := "test"
    varValueVarsStructure := "test"
    var varValueVarsStructureInterface interface{} = varValueVarsStructure
    var listVarsStructurebody = []model.VarsStructure{
        {
            VarKey: &varKeyVarsStructure,
            VarValue: &varValueVarsStructureInterface,
        },
    }
    templateKeyConformancePackRequestBody := "example-001.yaml"
    agencyNameConformancePackRequestBody := "config_conformance_pack_template"
    request.Body = &model.ConformancePackRequestBody{
        VarsStructure: &listVarsStructurebody,
        TemplateKey: &templateKeyConformancePackRequestBody,
        AgencyName: &agencyNameConformancePackRequestBody,
        Name: "conformance_pack_name",
    }
    response, err := client.CreateConformancePack(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.3 Querying a Conformance Package

Function

This API is used to query a conformance package by ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}

Table 4-609 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Request Parameters

Table 4-610 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-611 Response body parameters

Parameter	Type	Description
id	String	Specifies the ID of a conformance package.
name	String	Specifies the conformance package name.
stack_id	String	Specifies the unique ID of a resource stack. Minimum: 1 Maximum: 64
stack_name	String	Specifies the name of a resource stack. Minimum: 1 Maximum: 64
deployment_id	String	Specifies the deployment ID.
created_at	String	Specifies the time when a conformance package was created.
updated_at	String	Specifies the time when a conformance package was updated.
template_key	String	The name of a predefined conformance package template.
template_uri	String	The OBS address of a conformance package template
agency_name	String	The name of an agency.
status	String	Specifies the deployment status of a conformance package.

Parameter	Type	Description
error_message	String	Specifies the error message when you failed to deploy or delete a conformance package.
vars_structure	Array of VarsStructure objects	Specifies the parameters of a conformance package.
created_by	String	Creator

Table 4-612 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Status code: 400

Table 4-613 Response body parameters

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

Status code: 403

Table 4-614 Response body parameters

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

Status code: 404

Table 4-615 Response body parameters

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

Status code: 500

Table 4-616 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "a0123456789",
  "name" : "conformance_pack_name-e35r",
  "stack_id" : "stack-id-1234",
  "stack_name" : "stack-name-1234",
  "deployment_id" : "deployment_id_1234",
  "created_at" : "2023-02-28T09:39:19Z",
  "updated_at" : "2023-02-28T09:39:19Z",
  "status" : "CREATE_FAILED",
  "error_message" : "policy assignments exceed quota",
  "vars_structure" : [ {
    "var_key" : "test",
    "var_value" : "test"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowConformancePackRequest request = new ShowConformancePackRequest();
        try {
            ShowConformancePackResponse response = client.showConformancePack(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowConformancePackRequest()
```

```
response = client.show_conformance_pack(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowConformancePackRequest{}
    response, err := client.ShowConformancePack(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.

Status Code	Description
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.4 Deleting a Conformance Package

Function

This API is used to delete a conformance package.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}

Table 4-617 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Request Parameters

Table 4-618 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 400

Table 4-619 Response body parameters

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

Status code: 403

Table 4-620 Response body parameters

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

Status code: 404

Table 4-621 Response body parameters

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

Status code: 500

Table 4-622 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class DeleteConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteConformancePackRequest request = new DeleteConformancePackRequest();
        try {
            DeleteConformancePackResponse response = client.deleteConformancePack(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteConformancePackRequest()
        response = client.delete_conformance_pack(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()
```

```

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.DeleteConformancePackRequest{}
response, err := client.DeleteConformancePack(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Operation accepted.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.5 Updating Conformance Packages

Function

Updating conformance packages

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}

Table 4-623 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Request Parameters

Table 4-624 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	The name of a conformance package. Minimum: 1 Maximum: 64
vars_structure	No	Array of VarsStructure objects	Parameters of a conformance package. Array Length: 0 - 50

Table 4-625 VarsStructure

Parameter	Mandatory	Type	Description
var_key	No	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	No	Object	Specifies the value of a parameter.

Response Parameters

Status code: **200**

Table 4-626 Response body parameters

Parameter	Type	Description
id	String	Specifies the ID of a conformance package.
name	String	Specifies the conformance package name.
stack_id	String	Specifies the unique ID of a resource stack. Minimum: 1 Maximum: 64
stack_name	String	Specifies the name of a resource stack. Minimum: 1 Maximum: 64
deployment_id	String	Specifies the deployment ID.
created_at	String	Specifies the time when a conformance package was created.
updated_at	String	Specifies the time when a conformance package was updated.
template_key	String	The name of a predefined conformance package template.
template_uri	String	The OBS address of a conformance package template
agency_name	String	The name of an agency.
status	String	Specifies the deployment status of a conformance package.
error_message	String	Specifies the error message when you failed to deploy or delete a conformance package.
vars_structure	Array of VarsStructure objects	Specifies the parameters of a conformance package.
created_by	String	Creator

Table 4-627 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Status code: 400

Table 4-628 Response body parameters

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

Status code: 403

Table 4-629 Response body parameters

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

Status code: 404

Table 4-630 Response body parameters

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

Status code: 500

Table 4-631 Response body parameters

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

Example Requests

Updating conformance packages

```
PUT https://{endpoint}/v1/resource-manager/domains/{domain_id}/conformance-packs/
{conformance_pack_id}
{
  "name" : "conformance_pack_name-e35r",
```

```
"vars_structure" : [ {  
  "var_key" : "test",  
  "var_value" : "test"  
} ]  
}
```

Example Responses

Status code: 200

Updated.

```
{  
  "id" : "a0123456789",  
  "name" : "conformance_pack_name-e35r",  
  "stack_id" : "stack-id-1234",  
  "stack_name" : "stack-name-1234",  
  "deployment_id" : "deployment_id_1234",  
  "created_at" : "2023-02-28T09:39:19Z",  
  "updated_at" : "2023-02-28T09:40:19Z",  
  "status" : "UPDATE_IN_PROGRESS",  
  "error_message" : "policy assignments exceed quota",  
  "vars_structure" : [ {  
    "var_key" : "test",  
    "var_value" : "test"  
  } ]  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Updating conformance packages

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
import java.util.List;  
import java.util.ArrayList;  
  
public class UpdateConformancePackSolution {  
  
  public static void main(String[] args) {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    // environment variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running  
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    String ak = System.getenv("CLOUD_SDK_AK");  
    String sk = System.getenv("CLOUD_SDK_SK");  
  
    ICredential auth = new GlobalCredentials()  
      .withAk(ak)  
      .withSk(sk);  
  
    ConfigClient client = ConfigClient.newBuilder()
```



```
        .withCredential(auth)
        .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
        .build();
UpdateConformancePackRequest request = new UpdateConformancePackRequest();
UpdateConformancePackRequestBody body = new UpdateConformancePackRequestBody();
List<VarsStructure> listbodyVarsStructure = new ArrayList<>();
listbodyVarsStructure.add(
    new VarsStructure()
        .withVarKey("test")
        .withVarValue("test")
);
body.withVarsStructure(listbodyVarsStructure);
body.withName("conformance_pack_name-e35r");
request.withBody(body);
try {
    UpdateConformancePackResponse response = client.updateConformancePack(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Updating conformance packages

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateConformancePackRequest()
        listVarsStructurebody = [
            VarsStructure(
                var_key="test",
                var_value="test"
            )
        ]
        request.body = UpdateConformancePackRequestBody(
            vars_structure=listVarsStructurebody,
            name="conformance_pack_name-e35r"
```

```
)
response = client.update_conformance_pack(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Updating conformance packages

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateConformancePackRequest{}
    varKeyVarsStructure:= "test"
    varValueVarsStructure:= "test"
    var varValueVarsStructureInterface interface{} = varValueVarsStructure
    var listVarsStructurebody = []model.VarsStructure{
        {
            VarKey: &varKeyVarsStructure,
            VarValue: &varValueVarsStructureInterface,
        },
    }
    request.Body = &model.UpdateConformancePackRequestBody{
        VarsStructure: &listVarsStructurebody,
        Name: "conformance_pack_name-e35r",
    }
    response, err := client.UpdateConformancePack(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Updated.
400	Invalid parameter.
403	Authentication failed.
404	Conformance package not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.9.6 Querying the Compliance Results of Conformance Packages

Function

This API is used to query the compliance results of conformance packages.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/conformance-packs/compliance/summary

Table 4-632 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-633 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
conformance_pack_name	No	String	Specifies the conformance package name. Minimum: 1 Maximum: 64

Request Parameters

Table 4-634 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-635 Response body parameters

Parameter	Type	Description
value	Array of ConformancePackComplianceSummary objects	Specifies the summary of the compliance results of conformance packages.

Parameter	Type	Description
page_info	PageInfo object	Specifies the pagination object.

Table 4-636 ConformancePackComplianceSummary

Parameter	Type	Description
id	String	Specifies the ID of a conformance package.
name	String	Specifies the conformance package name.
compliance	String	Specifies the compliance result of a conformance package.

Table 4-637 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-638 Response body parameters

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

Status code: 403

Table 4-639 Response body parameters

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

Status code: 500

Table 4-640 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "id": "a0123456789",
    "name": "conformance_pack_name-e35r",
    "compliance": "NonCompliant"
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;
```

```
public class CollectConformancePackComplianceSummarySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CollectConformancePackComplianceSummaryRequest request = new
        CollectConformancePackComplianceSummaryRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        request.withConformancePackName("<conformance_pack_name>");
        try {
            CollectConformancePackComplianceSummaryResponse response =
            client.collectConformancePackComplianceSummary(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
```

```
request = CollectConformancePackComplianceSummaryRequest()
request.limit = <limit>
request.marker = "<marker>"
request.conformance_pack_name = "<conformance_pack_name>"
response = client.collect_conformance_pack_compliance_summary(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CollectConformancePackComplianceSummaryRequest{
        limitRequest:= int32(<limit>)
        request.Limit = &limitRequest
        markerRequest:= "<marker>"
        request.Marker = &markerRequest
        conformancePackNameRequest:= "<conformance_pack_name>"
        request.ConformancePackName = &conformancePackNameRequest
        response, err := client.CollectConformancePackComplianceSummary(request)
        if err == nil {
            fmt.Printf("%+v\n", response)
        } else {
            fmt.Println(err)
        }
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.7 Querying the Compliance Results of Rules in a Conformance Package

Function

This API is used to query the compliance results of rules in a conformance package.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}/compliance

Table 4-641 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Table 4-642 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
policy_assignment_name	No	String	Specifies the rule name. Minimum: 1 Maximum: 64

Request Parameters

Table 4-643 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-644 Response body parameters

Parameter	Type	Description
value	Array of ConformancePackCompliance objects	Specifies the compliance results of rules in a conformance package.
page_info	PageInfo object	Specifies the pagination object.

Table 4-645 ConformancePackCompliance

Parameter	Type	Description
policy_assignment_id	String	Specifies the rule ID.
policy_assignment_name	String	Specifies the rule name.
policy_assignment_compliance	String	Specifies the compliance result of a rule.

Table 4-646 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-647** Response body parameters

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

Status code: 403**Table 4-648** Response body parameters

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

Status code: 404

Table 4-649 Response body parameters

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

Status code: 500

Table 4-650 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "policy_assignment_id": "p0123456789",
    "policy_assignment_name": "policy_assignment_name-0709",
    "policy_assignment_compliance": "NonCompliant"
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
```

```
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListConformancePackComplianceByPackIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListConformancePackComplianceByPackIdRequest request = new
        ListConformancePackComplianceByPackIdRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        request.withPolicyAssignmentName("<policy_assignment_name>");
        try {
            ListConformancePackComplianceByPackIdResponse response =
            client.listConformancePackComplianceByPackId(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
```

```
.build()

try:
    request = ListConformancePackComplianceByPackIdRequest()
    request.limit = <limit>
    request.marker = "<marker>"
    request.policy_assignment_name = "<policy_assignment_name>"
    response = client.list_conformance_pack_compliance_by_pack_id(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListConformancePackComplianceByPackIdRequest{}
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    policyAssignmentNameRequest := "<policy_assignment_name>"
    request.PolicyAssignmentName = &policyAssignmentNameRequest
    response, err := client.ListConformancePackComplianceByPackId(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.8 Querying the Details about Compliance Results of Resources Evaluated by Rules in a Conformance Package

Function

This API is used to query the details about compliance results of resources evaluated by rules in a conformance package.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}/compliance/details

Table 4-651 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Table 4-652 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
policy_assignment_name	No	String	Specifies the rule name. Minimum: 1 Maximum: 64

Request Parameters

Table 4-653 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: **200**

Table 4-654 Response body parameters

Parameter	Type	Description
value	Array of ConformancePackComplianceDetail objects	Specifies the details about compliance results of rules in a conformance package.
page_info	PageInfo object	Specifies the pagination object.

Table 4-655 ConformancePackComplianceDetail

Parameter	Type	Description
policy_assignment_id	String	Specifies the assignment ID.
policy_assignment_name	String	Specifies the rule name.
resource_id	String	Specifies the ID of a resource to be evaluated.
compliance_state	String	Specifies the compliance result of a rule.
evaluation_time	String	Specifies the time for evaluating resources.

Table 4-656 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-657 Response body parameters

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

Status code: 403

Table 4-658 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 404**Table 4-659** Response body parameters

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

Status code: 500**Table 4-660** Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "policy_assignment_id": "p0123456789",
    "policy_assignment_name": "policy_assignment_name-0709",
    "resource_id": "ecs_001",
    "compliance_state": "NonCompliant",
    "evaluation_time": 1667378537566
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListConformancePackComplianceDetailsByPackIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListConformancePackComplianceDetailsByPackIdRequest request = new
        ListConformancePackComplianceDetailsByPackIdRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        request.withPolicyAssignmentName("<policy_assignment_name>");
        try {
            ListConformancePackComplianceDetailsByPackIdResponse response =
            client.listConformancePackComplianceDetailsByPackId(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = GlobalCredentials(ak, sk) \

client = ConfigClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListConformancePackComplianceDetailsByPackIdRequest()
    request.limit = <limit>
    request.marker = "<marker>"
    request.policy_assignment_name = "<policy_assignment_name>"
    response = client.list_conformance_pack_compliance_details_by_pack_id(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListConformancePackComplianceDetailsByPackIdRequest{}
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    policyAssignmentNameRequest:= "<policy_assignment_name>"
    request.PolicyAssignmentName = &policyAssignmentNameRequest
    response, err := client.ListConformancePackComplianceDetailsByPackId(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.9 Querying Scores of Conformance Packages

Function

This API is used to query scores of conformance packages.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/domains/{domain_id}/conformance-packs/scores

Table 4-661 Path Parameters

Parameter	Mandatory	Type	Description
domain_id	Yes	String	Specifies tags. Maximum: 36

Table 4-662 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
conformance_pack_name	No	String	Specifies the conformance package name. Minimum: 1 Maximum: 64

Request Parameters

Table 4-663 Request header parameters

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-664 Response body parameters

Parameter	Type	Description
value	Array of ConformancePackScore objects	Specifies the scores of conformance packages.
page_info	PageInfo object	Specifies the pagination object.

Table 4-665 ConformancePackScore

Parameter	Type	Description
id	String	Specifies the ID of a conformance package.
name	String	Specifies the conformance package name.
score	String	Specifies the score of a conformance package.

Table 4-666 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-667 Response body parameters

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

Status code: 403

Table 4-668 Response body parameters

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

Status code: 404

Table 4-669 Response body parameters

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

Status code: 500

Table 4-670 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "id": "a0123456789",
    "name": "conformance_pack_name-e35r",
    "score": "98.20"
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;
```



```
public class ListConformancePackComplianceScoresSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListConformancePackComplianceScoresRequest request = new
        ListConformancePackComplianceScoresRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        request.withConformancePackName("<conformance_pack_name>");
        try {
            ListConformancePackComplianceScoresResponse response =
            client.listConformancePackComplianceScores(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
```

```
request = ListConformancePackComplianceScoresRequest()
request.limit = <limit>
request.marker = "<marker>"
request.conformance_pack_name = "<conformance_pack_name>"
response = client.list_conformance_pack_compliance_scores(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListConformancePackComplianceScoresRequest{}
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    markerRequest := "<marker>"
    request.Marker = &markerRequest
    conformancePackNameRequest := "<conformance_pack_name>"
    request.ConformancePackName = &conformancePackNameRequest
    response, err := client.ListConformancePackComplianceScores(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.10 Querying Built-in Conformance Package Templates

Function

This API is used to query built-in conformance package templates.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/conformance-packs/templates

Table 4-671 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Parameter	Mandatory	Type	Description
template_key	No	String	Specifies the name of a built-in conformance package template. Minimum: 1 Maximum: 128

Request Parameters

Table 4-672 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Specifies the language of the information to return. The default name is zh-cn , which can be zh-cn and en-us . Default: zh-cn
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-673 Response body parameters

Parameter	Type	Description
value	Array of ConformancePackTemplate objects	Specifies built-in conformance package templates.
page_info	PageInfo object	Specifies the pagination object.

Table 4-674 ConformancePackTemplate

Parameter	Type	Description
id	String	Specifies the ID of a built-in conformance package template.
template_key	String	Specifies the name of a built-in conformance package template.
description	String	Specifies the description for a built-in conformance package template.
template_body	String	Specifies the content of a built-in conformance package template.
parameters	Map<String, TemplateParameterDefinition>	Specifies the parameters for a built-in conformance package template.

Table 4-675 TemplateParameterDefinition

Parameter	Type	Description
name	String	Specifies the name of a parameter for a built-in conformance package template.
description	String	Specifies the description of a parameter for a built-in conformance package template.
default_value	Object	Specifies the default value of a parameter for a built-in conformance package.
allowed_values	Array of objects	Specifies the list of allowed parameter values.
minimum	Float	Specifies the minimum value of the policy parameter.
maximum	Float	Specifies the maximum value of the policy parameter.
min_items	Integer	Specifies the minimum number of the policy parameter.
max_items	Integer	Specifies the maximum number of the policy parameter.
min_length	Integer	Specifies the minimum string length for policy parameters or for each item.
max_length	Integer	Specifies the maximum string length for policy parameters or for each item.

Parameter	Type	Description
pattern	String	Specifies the pattern for policy parameters or for each item.
type	String	Specifies the type of a parameter for a built-in conformance package template.

Table 4-676 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-677** Response body parameters

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

Status code: 403**Table 4-678** Response body parameters

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

Status code: 500

Table 4-679 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "value": [ {
    "id": "k0123456789",
    "template_key": "example-001.yaml",
    "description": "Example of built-in conformance package templates",
    "template_body": "Omit it.",
    "parameters": {
      "listOfAllowedFlavors": {
        "description": "The list of allowed flavor types.",
        "type": "Array",
        "default_value": "STAT",
        "allowed_values": [ "STAT" ]
      }
    }
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListBuiltInConformancePackTemplatesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```

```
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ListBuiltInConformancePackTemplatesRequest request = new
ListBuiltInConformancePackTemplatesRequest();
request.withLimit(<limit>);
request.withMarker("<marker>");
request.withTemplateKey("<template_key>");
try {
    ListBuiltInConformancePackTemplatesResponse response =
client.listBuiltInConformancePackTemplates(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListBuiltInConformancePackTemplatesRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.template_key = "<template_key>"
        response = client.list_built_in_conformance_pack_templates(request)
        print(response)
```



```
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListBuiltInConformancePackTemplatesRequest{}
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    markerRequest:= "<marker>"
    request.Marker = &markerRequest
    templateKeyRequest:= "<template_key>"
    request.TemplateKey = &templateKeyRequest
    response, err := client.ListBuiltInConformancePackTemplates(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.

Status Code	Description
400	Invalid parameters.
403	User authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.11 Querying a Built-in Conformance Package Template

Function

This API is used to query a built-in conformance package template by ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/conformance-packs/templates/{template_id}

Table 4-680 Path Parameters

Parameter	Mandatory	Type	Description
template_id	Yes	String	Specifies the ID of a conformance package template. Maximum: 36

Request Parameters

Table 4-681 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Specifies the language of the information to return. The default name is zh-cn , which can be zh-cn and en-us . Default: zh-cn

Parameter	Mandatory	Type	Description
X-Security-Token	No	String	Security token (session token) for temporary security credentials. This parameter is mandatory when you make an API call using temporary security credentials.

Response Parameters

Status code: 200

Table 4-682 Response body parameters

Parameter	Type	Description
id	String	Specifies the ID of a built-in conformance package template.
template_key	String	Specifies the name of a built-in conformance package template.
description	String	Specifies the description for a built-in conformance package template.
template_body	String	Specifies the content of a built-in conformance package template.
parameters	Map<String, TemplateParameterDefinition>	Specifies the parameters for a built-in conformance package template.

Table 4-683 TemplateParameterDefinition

Parameter	Type	Description
name	String	Specifies the name of a parameter for a built-in conformance package template.
description	String	Specifies the description of a parameter for a built-in conformance package template.
default_value	Object	Specifies the default value of a parameter for a built-in conformance package.
allowed_values	Array of objects	Specifies the list of allowed parameter values.
minimum	Float	Specifies the minimum value of the policy parameter.

Parameter	Type	Description
maximum	Float	Specifies the maximum value of the policy parameter.
min_items	Integer	Specifies the minimum number of the policy parameter.
max_items	Integer	Specifies the maximum number of the policy parameter.
min_length	Integer	Specifies the minimum string length for policy parameters or for each item.
max_length	Integer	Specifies the maximum string length for policy parameters or for each item.
pattern	String	Specifies the pattern for policy parameters or for each item.
type	String	Specifies the type of a parameter for a built-in conformance package template.

Status code: 400**Table 4-684** Response body parameters

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

Status code: 403**Table 4-685** Response body parameters

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

Status code: 404

Table 4-686 Response body parameters

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

Status code: 500

Table 4-687 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "id" : "k0123456789",
  "template_key" : "example-001.yaml",
  "description" : "Example of a built-in conformance package template",
  "template_body" : "Omit it.",
  "parameters" : {
    "listOfAllowedFlavors" : {
      "description" : "The list of allowed flavor types.",
      "type" : "Array",
      "default_value" : "STAT",
      "allowed_values" : [ "STAT" ]
    }
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
```

```
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowBuiltInConformancePackTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowBuiltInConformancePackTemplateRequest request = new
        ShowBuiltInConformancePackTemplateRequest();
        try {
            ShowBuiltInConformancePackTemplateResponse response =
            client.showBuiltInConformancePackTemplate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
```

```

request = ShowBuiltInConformancePackTemplateRequest()
response = client.show_built_in_conformance_pack_template(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)

```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowBuiltInConformancePackTemplateRequest{}
    response, err := client.ShowBuiltInConformancePackTemplate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameters.

Status Code	Description
403	User authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.12 Create a conformance package.

Function

Create a new organization conformance package.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/organizations/{organization_id}/conformance-packs

Table 4-688 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Request Parameters

Table 4-689 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Organization conformance package name Minimum: 1 Maximum: 64
excluded_accounts	No	Array of strings	The excluded accounts for conformance package deployment

Parameter	Mandatory	Type	Description
template_key	No	String	The name of a predefined conformance package Maximum: 128
template_body	No	String	The content of a custom conformance package Minimum: 0 Maximum: 51200
template_uri	No	String	The OBS address of a conformance package Minimum: 0 Maximum: 1024
vars_structure	No	Array of VarsStructure objects	Parameters of a conformance package Array Length: 0 - 50

Table 4-690 VarsStructure

Parameter	Mandatory	Type	Description
var_key	No	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	No	Object	Specifies the value of a parameter.

Response Parameters

Status code: 200

Table 4-691 Response body parameters

Parameter	Type	Description
org_conformance_pack_id	String	Organization conformance package ID
org_conformance_pack_name	String	Organization conformance package name

Parameter	Type	Description
owner_id	String	The creator of an organization conformance package
organization_id	String	Organization ID
org_conformance_package_urn	String	The unique identifier of resources in an organization conformance package
vars_structure	Array of VarsStructure objects	Parameters of a conformance package
created_at	String	The creation time of an organization conformance package
updated_at	String	The update time of an organization conformance package

Table 4-692 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Status code: 400

Table 4-693 Response body parameters

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

Status code: 403

Table 4-694 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 500

Table 4-695 Response body parameters

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

Example Requests

```
POST https://{endpoint}/v1/resource-manager/organizations/{organization_id}/conformance-packs
{
  "name" : "org_conformance_pack_name-e35r",
  "template_key" : "example-001.yaml",
  "template_body" : null,
  "template_uri" : null,
  "vars_structure" : null
}
```

Example Responses

Status code: 200

Operation succeeded.

```
{
  "org_conformance_pack_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
  "org_conformance_pack_name" : "org_conformance_pack_name-e35r",
  "owner_id" : "a1b2c043fab784a45ad88f5ef6a4bdaas",
  "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
  "org_conformance_pack_urn" :
  "rms:a1b2c043fab784a45ad88f5ef6a4bdaas:organizationConformancePack:o-doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
  "created_at" : "2023-02-28T09:39:19Z",
  "updated_at" : "2023-02-28T09:39:19Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
```

```
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class CreateOrganizationConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateOrganizationConformancePackRequest request = new
        CreateOrganizationConformancePackRequest();
        OrgConformancePackRequestBody body = new OrgConformancePackRequestBody();
        body.withTemplateKey("example-001.yaml");
        body.withName("org_conformance_pack_name-e35r");
        request.withBody(body);
        try {
            CreateOrganizationConformancePackResponse response =
            client.createOrganizationConformancePack(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
```

```
.with_credentials(credentials) \  
.with_region(ConfigRegion.value_of("<YOUR REGION>")) \  
.build()  
  
try:  
    request = CreateOrganizationConformancePackRequest()  
    request.body = OrgConformancePackRequestBody(  
        template_key="example-001.yaml",  
        name="org_conformance_pack_name-e35r"  
    )  
    response = client.create_organization_conformance_pack(request)  
    print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"  
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := global.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := config.NewConfigClient(  
        config.ConfigClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.CreateOrganizationConformancePackRequest{}  
    templateKeyOrgConformancePackRequestBody := "example-001.yaml"  
    request.Body = &model.OrgConformancePackRequestBody{  
        TemplateKey: &templateKeyOrgConformancePackRequestBody,  
        Name: "org_conformance_pack_name-e35r",  
    }  
    response, err := client.CreateOrganizationConformancePack(request)  
    if err == nil {  
        fmt.Printf("%v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	Authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.13 List organization conformance packages.

Function

List organization conformance packages of a user.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/conformance-packs

Table 4-696 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Table 4-697 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200

Parameter	Mandatory	Type	Description
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
organization_conformance_pack_id	No	String	Organization conformance package ID Maximum: 36
conformance_pack_name	No	String	Specifies the conformance package name. Minimum: 1 Maximum: 64

Request Parameters

None

Response Parameters

Status code: 200

Table 4-698 Response body parameters

Parameter	Type	Description
organization_conformance_packs	Array of OrgConformancePackResponse objects	Organization conformance package list
page_info	PageInfo object	Specifies the pagination object.

Table 4-699 OrgConformancePackResponse

Parameter	Type	Description
org_conformance_pack_id	String	Organization conformance package ID
org_conformance_pack_name	String	Organization conformance package name
owner_id	String	The creator of an organization conformance package

Parameter	Type	Description
organization_id	String	Organization ID
org_conformance_package_urn	String	The unique identifier of resources in an organization conformance package
vars_structure	Array of VarsStructure objects	Parameters of a conformance package
created_at	String	The creation time of an organization conformance package
updated_at	String	The update time of an organization conformance package

Table 4-700 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Table 4-701 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-702 Response body parameters

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

Status code: 403

Table 4-703 Response body parameters

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

Status code: 500

Table 4-704 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "organization_conformance_packs" : [ {
    "org_conformance_pack_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
    "org_conformance_pack_name" : "org_conformance_pack_name-e35r",
    "owner_id" : "a1b2c043fab784a45ad88f5ef6a4bdaas",
    "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
    "org_conformance_pack_urn" :
"rms::a1b2c043fab784a45ad88f5ef6a4bdaas:organizationConformancePack:o-
doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
    "created_at" : "2023-02-28T09:39:19Z",
    "updated_at" : "2023-02-28T09:39:19Z"
  } ],
  "page_info" : {
    "current_count" : 1,
    "next_marker" : null
  }
}
```

```
}  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.GlobalCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;  
import com.huaweicloud.sdk.config.v1.*;  
import com.huaweicloud.sdk.config.v1.model.*;  
  
public class ListOrganizationConformancePacksSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new GlobalCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ConfigClient client = ConfigClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListOrganizationConformancePacksRequest request = new  
        ListOrganizationConformancePacksRequest();  
        request.withLimit(<limit>);  
        request.withMarker("<marker>");  
        request.withOrganizationConformancePackId("<organization_conformance_pack_id>");  
        request.withConformancePackName("<conformance_pack_name>");  
        try {  
            ListOrganizationConformancePacksResponse response =  
            client.listOrganizationConformancePacks(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8
```

```
from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListOrganizationConformancePacksRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.organization_conformance_pack_id = "<organization_conformance_pack_id>"
        request.conformance_pack_name = "<conformance_pack_name>"
        response = client.list_organization_conformance_packs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListOrganizationConformancePacksRequest{}
    limitRequest := int32(<limit>)
```

```

request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
organizationConformancePackIdRequest:= "<organization_conformance_pack_id>"
request.OrganizationConformancePackId = &organizationConformancePackIdRequest
conformancePackNameRequest:= "<conformance_pack_name>"
request.ConformancePackName = &conformancePackNameRequest
response, err := client.ListOrganizationConformancePacks(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	Authentication failed.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.14 View organization conformance packages.

Function

Obtain details of an organization conformance package based on an ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/conformance-packs/{conformance_pack_id}

Table 4-705 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Request Parameters

None

Response Parameters

Status code: 200

Table 4-706 Response body parameters

Parameter	Type	Description
org_conformance_pack_id	String	Organization conformance package ID
org_conformance_pack_name	String	Organization conformance package name
owner_id	String	The creator of an organization conformance package
organization_id	String	Organization ID
org_conformance_pack_urn	String	The unique identifier of resources in an organization conformance package
excluded_accounts	Array of strings	Excluded accounts for conformance package deployment
vars_structure	Array of VarsStructure objects	Parameters of a conformance package
created_at	String	The creation time of an organization conformance package
updated_at	String	The update time of an organization conformance package
template_key	String	The name of a predefined conformance package template.

Parameter	Type	Description
template_uri	String	The OBS address of a conformance package template

Table 4-707 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Status code: 400

Table 4-708 Response body parameters

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

Status code: 403

Table 4-709 Response body parameters

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

Status code: 404

Table 4-710 Response body parameters

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

Status code: 500

Table 4-711 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "org_conformance_pack_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
  "org_conformance_pack_name" : "org_conformance_pack_name-e35r",
  "owner_id" : "a1b2c043fab784a45ad88f5ef6a4bdaas",
  "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrknp",
  "org_conformance_pack_urn" :
  "rms::a1b2c043fab784a45ad88f5ef6a4bdaas:organizationConformancePack:o-
  doxbpkzqsdd51ti1k27w2y8fitakrknp/d02b7fa9d5a74e638c1402d0868f71fd",
  "created_at" : "2023-02-28T09:39:19Z",
  "updated_at" : "2023-02-28T09:39:19Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowOrganizationConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```

```
ICredential auth = new GlobalCredentials()
    .withAk(ak)
    .withSk(sk);

ConfigClient client = ConfigClient.newBuilder()
    .withCredential(auth)
    .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
    .build();
ShowOrganizationConformancePackRequest request = new
ShowOrganizationConformancePackRequest();
try {
    ShowOrganizationConformancePackResponse response =
client.showOrganizationConformancePack(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowOrganizationConformancePackRequest()
        response = client.show_organization_conformance_pack(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```



```

"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowOrganizationConformancePackRequest{}
    response, err := client.ShowOrganizationConformancePack(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	Authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.15 Delete organization conformance packages.

Function

Delete organization conformance packages of a user.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v1/resource-manager/organizations/{organization_id}/conformance-packs/{conformance_pack_id}

Table 4-712 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Request Parameters

None

Response Parameters

Status code: 400

Table 4-713 Response body parameters

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

Status code: 403

Table 4-714 Response body parameters

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

Status code: 404

Table 4-715 Response body parameters

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

Status code: 500

Table 4-716 Response body parameters

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

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;
```

```
public class DeleteOrganizationConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        DeleteOrganizationConformancePackRequest request = new
        DeleteOrganizationConformancePackRequest();
        try {
            DeleteOrganizationConformancePackResponse response =
            client.deleteOrganizationConformancePack(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteOrganizationConformancePackRequest()
        response = client.delete_organization_conformance_pack(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteOrganizationConformancePackRequest{}
    response, err := client.DeleteOrganizationConformancePack(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Operation accepted.
400	Invalid parameter.
403	Authentication failed.

Status Code	Description
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.16 Updating Organization Conformance Packages

Function

This API is used to update organization conformance packages.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v1/resource-manager/organizations/{organization_id}/conformance-packs/{conformance_pack_id}

Table 4-717 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34
conformance_pack_id	Yes	String	Specifies the ID of a conformance package. Maximum: 36

Request Parameters

Table 4-718 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	The name of an organization conformance package. Minimum: 1 Maximum: 60

Parameter	Mandatory	Type	Description
excluded_accounts	No	Array of strings	The accounts for which the conformance package will not be deployed.
vars_structure	No	Array of VarsStructure objects	Parameters of a conformance package. Array Length: 0 - 50

Table 4-719 VarsStructure

Parameter	Mandatory	Type	Description
var_key	No	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	No	Object	Specifies the value of a parameter.

Response Parameters

Status code: 200

Table 4-720 Response body parameters

Parameter	Type	Description
org_conformance_pack_id	String	Organization conformance package ID
org_conformance_pack_name	String	Organization conformance package name
owner_id	String	The creator of an organization conformance package
organization_id	String	Organization ID
org_conformance_pack_urn	String	The unique identifier of resources in an organization conformance package
vars_structure	Array of VarsStructure objects	Parameters of a conformance package

Parameter	Type	Description
created_at	String	The creation time of an organization conformance package
updated_at	String	The update time of an organization conformance package

Table 4-721 VarsStructure

Parameter	Type	Description
var_key	String	Specifies the name of a parameter. Minimum: 1 Maximum: 64
var_value	Object	Specifies the value of a parameter.

Status code: 400

Table 4-722 Response body parameters

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

Status code: 403

Table 4-723 Response body parameters

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

Status code: 404

Table 4-724 Response body parameters

Parameter	Type	Description
error_code	String	Specifies the error code.

Parameter	Type	Description
error_msg	String	Specifies the error message.

Status code: 500

Table 4-725 Response body parameters

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

Example Requests

Updating Organization Conformance Packages

POST https://{endpoint}/v1/resource-manager/organizations/{organization_id}/conformance-packs

```
{
  "name" : "org_conformance_pack_name-e35r",
  "vars_structure" : [ {
    "var_key" : "test",
    "var_value" : "test"
  } ]
}
```

Example Responses

Status code: 200

Update task created.

```
{
  "org_conformance_pack_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
  "org_conformance_pack_name" : "org_conformance_pack_name-e35r",
  "owner_id" : "a1b2c043fab784a45ad88f5ef6a4bdaas",
  "organization_id" : "o-doxbpkzqsdd51ti1k27w2y8fitakrkn",
  "org_conformance_pack_urn" :
  "rms::a1b2c043fab784a45ad88f5ef6a4bdaas:organizationConformancePack:o-
  doxbpkzqsdd51ti1k27w2y8fitakrkn/d02b7fa9d5a74e638c1402d0868f71fd",
  "created_at" : "2023-02-28T09:39:19Z",
  "updated_at" : "2023-02-28T09:39:19Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Updating Organization Conformance Packages

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateOrganizationConformancePackSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdateOrganizationConformancePackRequest request = new
        UpdateOrganizationConformancePackRequest();
        UpdateOrgConformancePackRequestBody body = new UpdateOrgConformancePackRequestBody();
        List<VarsStructure> listbodyVarsStructure = new ArrayList<>();
        listbodyVarsStructure.add(
            new VarsStructure()
                .withVarKey("test")
                .withVarValue("test")
        );
        body.withVarsStructure(listbodyVarsStructure);
        body.withName("org_conformance_pack_name-e35r");
        request.withBody(body);
        try {
            UpdateOrganizationConformancePackResponse response =
            client.updateOrganizationConformancePack(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Updating Organization Conformance Packages

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateOrganizationConformancePackRequest()
        listVarsStructurebody = [
            VarsStructure(
                var_key="test",
                var_value="test"
            )
        ]
        request.body = UpdateOrgConformancePackRequestBody(
            vars_structure=listVarsStructurebody,
            name="org_conformance_pack_name-e35r"
        )
        response = client.update_organization_conformance_pack(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Updating Organization Conformance Packages

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := config.NewConfigClient(
        config.ConfigClientBuilder().
```

```

WithRegion(region.ValueOf("<YOUR REGION>")).
WithCredential(auth).
Build()

request := &model.UpdateOrganizationConformancePackRequest{}
varKeyVarsStructure:= "test"
varValueVarsStructure:= "test"
var varValueVarsStructureInterface interface{} = varValueVarsStructure
var listVarsStructurebody = []model.VarsStructure{
    {
        VarKey: &varKeyVarsStructure,
        VarValue: &varValueVarsStructureInterface,
    },
}
request.Body = &model.UpdateOrgConformancePackRequestBody{
    VarsStructure: &listVarsStructurebody,
    Name: "org_conformance_pack_name-e35r",
}
response, err := client.UpdateOrganizationConformancePack(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Update task created.
400	Invalid parameter.
403	Authentication failed.
404	Conformance package not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.9.17 View deployment status of organization conformance packages.

Function

View deployment status of a user's organization conformance packages.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/conformance-packs/statuses

Table 4-726 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Table 4-727 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400
organization_conformance_pack_id	No	String	Organization conformance package ID Maximum: 36
conformance_pack_name	No	String	Specifies the conformance package name. Minimum: 1 Maximum: 64

Request Parameters

None

Response Parameters

Status code: 200

Table 4-728 Response body parameters

Parameter	Type	Description
statuses	Array of OrgConformancePackStatus objects	Organization conformance package list
page_info	PageInfo object	Specifies the pagination object.

Table 4-729 OrgConformancePackStatus

Parameter	Type	Description
org_conformance_pack_id	String	Organization conformance package ID
org_conformance_pack_name	String	Organization conformance package name
state	String	Deployment status of an organization conformance package
error_message	String	An error message for deploying or deleting an organization conformance package
created_at	String	The creation time of an organization conformance package
updated_at	String	The update time of an organization conformance package

Table 4-730 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400

Table 4-731 Response body parameters

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

Status code: 403

Table 4-732 Response body parameters

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

Status code: 404

Table 4-733 Response body parameters

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

Status code: 500

Table 4-734 Response body parameters

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

Example Requests

None

Example Responses

Status code: 200

Operation succeeded.

```
{
  "statuses": [ {
```

```
"org_conformance_pack_id" : "d02b7fa9d5a74e638c1402d0868f71fd",
"org_conformance_pack_name" : "org_conformance_pack_name-e35r",
"state" : "CREATE_SUCCESSFUL",
"error_message" : null,
"created_at" : "2023-08-16T03:13:50Z",
"updated_at" : "2023-08-16T03:13:50Z"
}],
"page_info" : {
  "current_count" : 1,
  "next_marker" : null
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ListOrganizationConformancePackStatusesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ListOrganizationConformancePackStatusesRequest request = new
        ListOrganizationConformancePackStatusesRequest();
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        request.withOrganizationConformancePackId("<organization_conformance_pack_id>");
        request.withConformancePackName("<conformance_pack_name>");
        try {
            ListOrganizationConformancePackStatusesResponse response =
            client.listOrganizationConformancePackStatuses(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
        }
    }
}
```



```
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListOrganizationConformancePackStatusesRequest()
        request.limit = <limit>
        request.marker = "<marker>"
        request.organization_conformance_pack_id = "<organization_conformance_pack_id>"
        request.conformance_pack_name = "<conformance_pack_name>"
        response = client.list_organization_conformance_pack_statuses(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := global.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

```

Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListOrganizationConformancePackStatusesRequest{}
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
organizationConformancePackIdRequest:= "<organization_conformance_pack_id>"
request.OrganizationConformancePackId = &organizationConformancePackIdRequest
conformancePackNameRequest:= "<conformance_pack_name>"
request.ConformancePackName = &conformancePackNameRequest
response, err := client.ListOrganizationConformancePackStatuses(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
    
```

More

For SDK sample code of more programming languages, see the [Sample Code](#) tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	Authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.9.18 View deployment status of an organization conformance package.

Function

View deployment status of a user's specific organization conformance packages.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/organizations/{organization_id}/conformance-packs/
detailed-statuses

Table 4-735 Path Parameters

Parameter	Mandatory	Type	Description
organization_id	Yes	String	Specifies the organization ID. Maximum: 34

Table 4-736 Query Parameters

Parameter	Mandatory	Type	Description
conformance_pack_name	No	String	Conformance package name Minimum: 1 Maximum: 64
organization_conformance_pack_id	No	String	Organization conformance package ID Maximum: 36
domain_id	No	String	Specifies the account ID. Maximum: 36
state	No	String	Deployment status (case sensitive) Maximum: 20
limit	No	Integer	Specifies the maximum number of records to return. Minimum: 1 Maximum: 200 Default: 200
marker	No	String	Specifies the pagination parameter. Minimum: 4 Maximum: 400

Request Parameters

None

Response Parameters

Status code: 200

Table 4-737 Response body parameters

Parameter	Type	Description
statuses	Array of OrgConformancePackDetailedStatus objects	Organization conformance package list
page_info	PageInfo object	Specifies the pagination object.

Table 4-738 OrgConformancePackDetailedStatus

Parameter	Type	Description
domain_id	String	Specifies the ID of the member account.
conformance_pack_id	String	Conformance package ID
conformance_pack_name	String	Specifies the conformance package name.
state	String	Deployment status of a conformance package
error_message	String	Specifies the error message when you fail to deploy or delete an organization conformance package.
created_at	String	Creation time
updated_at	String	Update time

Table 4-739 PageInfo

Parameter	Type	Description
current_count	Integer	Specifies the resource quantity on the current page. Minimum: 0 Maximum: 200
next_marker	String	Specifies the marker value of the next page. Minimum: 4 Maximum: 400

Status code: 400**Table 4-740** Response body parameters

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

Status code: 403**Table 4-741** Response body parameters

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

Status code: 404**Table 4-742** Response body parameters

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

Status code: 500**Table 4-743** Response body parameters

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

Example Requests

None

Example Responses**Status code: 200**

Operation succeeded.

```
{
  "statuses": [ {
    "domain_id": "d02b7fa9d5a74e638c1402d0868f71fd",
    "conformance_pack_id": "d02b7fa9d5a74e638c1402d0868f71fd",
    "conformance_pack_name": "org_conformance_pack_name-e35r",
    "state": "CREATE_SUCCESSFUL",
    "error_message": "null",
    "created_at": "2023-01-17T09:38:26.165Z",
    "updated_at": "2023-01-17T09:38:26.165Z"
  } ],
  "page_info": {
    "current_count": 1,
    "next_marker": null
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.GlobalCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.config.v1.region.ConfigRegion;
import com.huaweicloud.sdk.config.v1.*;
import com.huaweicloud.sdk.config.v1.model.*;

public class ShowOrganizationConformancePackDetailedStatusesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new GlobalCredentials()
            .withAk(ak)
            .withSk(sk);

        ConfigClient client = ConfigClient.newBuilder()
            .withCredential(auth)
            .withRegion(ConfigRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowOrganizationConformancePackDetailedStatusesRequest request = new
        ShowOrganizationConformancePackDetailedStatusesRequest();
        request.withConformancePackName("<conformance_pack_name>");
        request.withOrganizationConformancePackId("<organization_conformance_pack_id>");

        request.withState(ShowOrganizationConformancePackDetailedStatusesRequest.StateEnum.fromValue("<stat
        e>"));
        request.withLimit(<limit>);
        request.withMarker("<marker>");
        try {
            ShowOrganizationConformancePackDetailedStatusesResponse response =
            client.showOrganizationConformancePackDetailedStatuses(request);
            System.out.println(response.toString());
        }
    }
}
```

```

    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
}

```

Python

```

# coding: utf-8

from huaweicloudsdkcore.auth.credentials import GlobalCredentials
from huaweicloudsdkconfig.v1.region.config_region import ConfigRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkconfig.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = GlobalCredentials(ak, sk) \

    client = ConfigClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ConfigRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowOrganizationConformancePackDetailedStatusesRequest()
        request.conformance_pack_name = "<conformance_pack_name>"
        request.organization_conformance_pack_id = "<organization_conformance_pack_id>"
        request.state = "<state>"
        request.limit = <limit>
        request.marker = "<marker>"
        response = client.show_organization_conformance_pack_detailed_statuses(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)

```

Go

```

package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/global"
    config "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/config/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment

```

```

variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := global.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := config.NewConfigClient(
    config.ConfigClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowOrganizationConformancePackDetailedStatusesRequest{}
conformancePackNameRequest:= "<conformance_pack_name>"
request.ConformancePackName = &conformancePackNameRequest
organizationConformancePackIdRequest:= "<organization_conformance_pack_id>"
request.OrganizationConformancePackId = &organizationConformancePackIdRequest
stateRequest:=
model.GetShowOrganizationConformancePackDetailedStatusesRequestStateEnum().<STATE>
request.State = &stateRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
markerRequest:= "<marker>"
request.Marker = &markerRequest
response, err := client.ShowOrganizationConformancePackDetailedStatuses(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}

```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Operation succeeded.
400	Invalid parameter.
403	Authentication failed.
404	Resources not found.
500	Server error.

Error Codes

See [Error Codes](#).

4.10 Resource Tags

4.10.1 Querying Resources

Function

This API is used to filter resources. TMS allows you to search for resources by tag. Query results will be displayed in a list. Up to 20 tags can be specified in parameters: tags, tags_any, not_tags, and not_tags_any.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/{resource_type}/resource-instances/filter

Table 4-744 Path Parameters

Parameter	Mandatory	Type	Description
resource_type	Yes	String	Resource type.

Table 4-745 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Up to how many records can be queried. This parameter is not available when the action element is set to count. If action is set to filter, the default value is 1000. The maximum value is 1,000, and the minimum value is 1. Minimum: 1 Maximum: 1000 Default: 1000

Parameter	Mandatory	Type	Description
offset	No	Integer	Index position. This parameter is not available when the action element is set to count. If the offset element is set to N, the resource query starts from the N+1 piece of data. If action is set to filter, the value of offset is 0 by default, indicating that the query starts from the first piece of data. The offset value must be a number and cannot be a negative number. Minimum: 0 Default: 0

Request Parameters

Table 4-746 Request body parameters

Parameter	Mandatory	Type	Description
without_any_tag	No	Boolean	Tags to be excluded. If this parameter is set to true, all resources that do not have the specified tags will be returned. If this parameter is set to false or is not specified, it does not take effect. Default: false

Parameter	Mandatory	Type	Description
tags	No	Array of Tag objects	Specified tags. Up to 20 tag keys can be specified, and up to 20 values can be specified for each key. Each tag value can be an empty array, but the tag structure cannot be missing. Each tag key must be unique, and the values of the same key must be unique. Resources containing any of the specified tags will be returned. Keys in this list are in an AND relationship while values in each key-value structure is in an OR relationship. If no tag filtering condition is specified, full data is returned.
matches	No	Array of Match objects	Search field. A field contains a key (such as resource_name) and a value. Keys are predefined and must be unique and valid. Whether fuzzy or exact match is used is determined based on the field key. For example, exact match is used for the field key resource_name. If the field value is an empty string, exact match is used. Since very few resources have an empty-string name, an empty list is returned. Exact match is used for the field key resource_id. Currently only resource_name is supported.

Table 4-747 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Tag key. A key can contain up to 128 Unicode characters and cannot be left blank. The system does not verify the character set of the key when searching for resources. The key element cannot be empty, an empty string, or spaces. Before using the key element, delete single-byte character (SBC) spaces before and after the value.</p> <p>Minimum: 1 Maximum: 128</p>
values	Yes	Array of strings	<p>Tag values. A key can contain up to 255 Unicode characters. Before verifying or using the value element, delete single-byte character (SBC) spaces before and after the value. The value can be an empty array but cannot be left blank. If no value is specified, any values are supported. Values are in an Or relationship. The system does not verify the character set of bvaluesb when searching for resources, but only verifies the length.</p> <p>Maximum: 255 Array Length: 0 - 20</p>

Table 4-748 Match

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Tag key. Currently, only the resource_name is supported.</p>
value	Yes	String	<p>Tag value. A tag value can contain up to 255 Unicode characters. The character set of a tag value will not be verified by the system.</p> <p>Maximum: 255</p>

Response Parameters

Status code: 200

Table 4-749 Response body parameters

Parameter	Type	Description
resources	Array of ResourceResp objects	
total_count	Integer	The total number of records.

Table 4-750 ResourceResp

Parameter	Type	Description
resource_id	String	Resource ID.
resource_detail	Object	Resource details. The field is reserved for extension and defaults to an empty string.
tags	Array of ResourceTag objects	Tags. If no tag is specified, this parameter is an empty array by default.
resource_name	String	Resource name. If no resource is specified, this parameter is an empty array by default. For EIP, IPs will be returned.

Table 4-751 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Status code: 400

Table 4-752 Response body parameters

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

Status code: 401

Table 4-753 Response body parameters

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

Status code: 403

Table 4-754 Response body parameters

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

Status code: 404

Table 4-755 Response body parameters

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

Status code: 500

Table 4-756 Response body parameters

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

Example Requests

Querying Instances by Tag

POST https://{endpoint}/v1/resource-manager/{resource_type}/resource-instances/filter

```
{
  "tags": [ {
    "key": "key1",
    "values": [ "value1", "value2" ]
  }, {
    "key": "key2",
    "values": [ "value1", "value2" ]
  } ],
  "matches": [ {
    "key": "resource_name",
    "value": "resource1"
  } ],
  "without_any_tag": "false"
}
```

Example Responses

Status code: 200

Succeeded.

```
{
  "resources": [ {
    "resource_detail": "",
    "resource_id": "cdfs_cefs_wesas_12_dsad",
    "resource_name": "resoucee1",
    "tags": [ {
      "key": "key1",
      "value": "value1"
    } ],
    "sys_tags": [ {
      "key": "_sys_enterprise_project_id",
      "value": "5aa119a8-d25b-45a7-8d1b-88e127885635"
    } ]
  } ],
  "total_count": 1000
}
```

Status Codes

Status Code	Description
200	Succeeded.
400	Invalid parameter.
401	Unauthorized.
403	Forbidden.
404	Resources not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.10.2 Querying Resource Statistics

Function

This API is used to query resource statistics. TMS allows you to filter resources by tag. Query results will be displayed in a list. Up to 20 tags can be specified in parameters: tags, tags_any, not_tags, and not_tags_any.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/{resource_type}/resource-instances/count

Table 4-757 Path Parameters

Parameter	Mandatory	Type	Description
resource_type	Yes	String	Resource type.

Request Parameters

Table 4-758 Request body parameters

Parameter	Mandatory	Type	Description
without_any_tag	No	Boolean	Tags to be excluded. If this parameter is set to true, all resources that do not have the specified tags will be returned. If this parameter is set to false or is not specified, it does not take effect. Default: false

Parameter	Mandatory	Type	Description
tags	No	Array of Tag objects	Specified tags. Up to 20 tag keys can be specified, and up to 20 values can be specified for each key. Each tag value can be an empty array, but the tag structure cannot be missing. Each tag key must be unique, and the values of the same key must be unique. Resources containing any of the specified tags will be returned. Keys in this list are in an AND relationship while values in each key-value structure is in an OR relationship. If no tag filtering condition is specified, full data is returned.
matches	No	Array of Match objects	Search field. A field contains a key (such as resource_name) and a value. Keys are predefined and must be unique and valid. Whether fuzzy or exact match is used is determined based on the field key. For example, exact match is used for the field key resource_name. If the field value is an empty string, exact match is used. Since very few resources have an empty-string name, an empty list is returned. Exact match is used for the field key resource_id. Currently only resource_name is supported.

Table 4-759 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Tag key. A key can contain up to 128 Unicode characters and cannot be left blank. The system does not verify the character set of the key when searching for resources. The key element cannot be empty, an empty string, or spaces. Before using the key element, delete single-byte character (SBC) spaces before and after the value.</p> <p>Minimum: 1 Maximum: 128</p>
values	Yes	Array of strings	<p>Tag values. A key can contain up to 255 Unicode characters. Before verifying or using the value element, delete single-byte character (SBC) spaces before and after the value. The value can be an empty array but cannot be left blank. If no value is specified, any values are supported. Values are in an Or relationship. The system does not verify the character set of bvaluesb when searching for resources, but only verifies the length.</p> <p>Maximum: 255 Array Length: 0 - 20</p>

Table 4-760 Match

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Tag key. Currently, only the resource_name is supported.</p>
value	Yes	String	<p>Tag value. A tag value can contain up to 255 Unicode characters. The character set of a tag value will not be verified by the system.</p> <p>Maximum: 255</p>

Response Parameters

Status code: 200

Table 4-761 Response body parameters

Parameter	Type	Description
total_count	Integer	The total number of records.

Status code: 400

Table 4-762 Response body parameters

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

Status code: 401

Table 4-763 Response body parameters

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

Status code: 403

Table 4-764 Response body parameters

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

Status code: 404

Table 4-765 Response body parameters

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

Status code: 500

Table 4-766 Response body parameters

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

Example Requests

Querying the Number of Instances By Tags

POST https://{endpoint}/v1/resource-manager/{resource_type}/resource-instances/count

```
{
  "tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  }, {
    "key" : "key2",
    "values" : [ "value1", "value2" ]
  } ],
  "matches" : [ {
    "key" : "resource_name",
    "value" : "resource1"
  } ]
}
```

Example Responses

Status code: 200

Succeeded.

```
{
  "total_count" : 1000
}
```

Status Codes

Status Code	Description
200	Succeeded.
400	Invalid parameter.

Status Code	Description
401	Unauthorized.
403	Forbidden.
404	Resources not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.10.3 Batch Adding Resource Tags

Function

This API is idempotent and is used to batch add or delete tags for a specified resource. You can add up to 20 tags to a resource.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/{resource_type}/{resource_id}/tags/create

Table 4-767 Path Parameters

Parameter	Mandatory	Type	Description
resource_type	Yes	String	Resource type.
resource_id	Yes	String	Resource ID.

Request Parameters

Table 4-768 Request body parameters

Parameter	Mandatory	Type	Description
tags	No	Array of ResourceTag objects	Tags. This parameter is mandatory when the tenant permission is used. For the op_service permission, choose either this field or bsys_tagsb. Array Length: 1 - 20

Table 4-769 ResourceTag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	Yes	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Response Parameters

Status code: 400

Table 4-770 Response body parameters

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

Status code: 401

Table 4-771 Response body parameters

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

Status code: 403

Table 4-772 Response body parameters

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

Status code: 404

Table 4-773 Response body parameters

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

Status code: 500

Table 4-774 Response body parameters

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

Example Requests

Adding Tags to Specified Instances in Batches

POST https://{endpoint}/v1/resource-manager/{resource_type}/{resource_id}/tags/create

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

```
"value" : "value2"  
}]  
}
```

Example Responses

None

Status Codes

Status Code	Description
204	No content.
400	Invalid parameter.
401	Unauthorized.
403	Forbidden.
404	Resources not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.10.4 Batch Deleting Resource Tags

Function

This API is idempotent and is used to batch add or delete tags for a specified resource. Each resource can have 20 tags at most.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v1/resource-manager/{resource_type}/{resource_id}/tags/delete

Table 4-775 Path Parameters

Parameter	Mandatory	Type	Description
resource_type	Yes	String	Resource type.
resource_id	Yes	String	Resource ID.

Request Parameters

Table 4-776 Request body parameters

Parameter	Mandatory	Type	Description
tags	No	Array of ResourceUnTag objects	Tags. This parameter is mandatory when the tenant permission is used. For the op_service permission, choose either this field or bsys_tagsb. Array Length: 1 - 20

Table 4-777 ResourceUnTag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	Yes	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Response Parameters

Status code: 400

Table 4-778 Response body parameters

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

Status code: 401

Table 4-779 Response body parameters

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

Status code: 403**Table 4-780** Response body parameters

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

Status code: 404**Table 4-781** Response body parameters

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

Status code: 500**Table 4-782** Response body parameters

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

Example Requests

Deleting Tags for Specified Instances in Batches

```
POST https://{endpoint}/v1/resource-manager/{resource_type}/{resource_id}/tags/delete
```

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

```
}]  
}
```

Example Responses

None

Status Codes

Status Code	Description
204	No content.
400	Invalid parameter.
401	Unauthorized.
403	Forbidden.
404	Resources not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.10.5 Querying Resource Tags

Function

This API is used to query resource tags.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/{resource_type}/{resource_id}/tags

Table 4-783 Path Parameters

Parameter	Mandatory	Type	Description
resource_type	Yes	String	Resource type.
resource_id	Yes	String	Resource ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 4-784 Response body parameters

Parameter	Type	Description
tags	Array of ResourceTag objects	Tags.

Table 4-785 ResourceTag

Parameter	Type	Description
key	String	Tag key. A tag key can contain up to 128 Unicode characters and must comply with the character set specifications in section 3.1. Minimum: 1 Maximum: 128
value	String	Tag value. A tag value can contain up to 255 Unicode characters and must comply with the character set specifications in section 3.2. Minimum: 0 Maximum: 255

Status code: 400

Table 4-786 Response body parameters

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

Status code: 401

Table 4-787 Response body parameters

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

Status code: 403

Table 4-788 Response body parameters

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

Status code: 404

Table 4-789 Response body parameters

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

Status code: 500

Table 4-790 Response body parameters

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

Example Requests

Querying Tags of a Specified Instance

```
GET https://{endpoint}/v1/resource-manager/{resource_type}/{resource_id}/tags
```

Example Responses

Status code: 200

Succeeded.

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

Status Codes

Status Code	Description
200	Succeeded.
400	Invalid parameter.
401	Unauthorized.
403	Forbidden.
404	Resources not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

4.10.6 Querying Resource Tags by Project

Function

This API is used to query tags of resources that are in a specified project.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v1/resource-manager/{resource_type}/tags

Table 4-791 Path Parameters

Parameter	Mandatory	Type	Description
resource_type	Yes	String	Resource type.

Table 4-792 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records that can be queried. The default value is 1000. The minimum value is 1. Minimum: 1 Maximum: 1000 Default: 1000
offset	No	Integer	Index position. If the offset element is set to N, the resource query starts from the N+1 piece of data. If action is set to filter, the value of offset is 0 by default, indicating that the query starts from the first piece of data. The offset value must be a number and cannot be a negative number. Minimum: 0 Default: 0

Request Parameters

None

Response Parameters

Status code: 200

Table 4-793 Response body parameters

Parameter	Type	Description
tags	Array of Tag objects	Tags.
total_count	Integer	The total number of records.

Table 4-794 Tag

Parameter	Type	Description
key	String	Tag key. A key can contain up to 128 Unicode characters and cannot be left blank. The system does not verify the character set of the key when searching for resources. The key element cannot be empty, an empty string, or spaces. Before using the key element, delete single-byte character (SBC) spaces before and after the value. Minimum: 1 Maximum: 128
values	Array of strings	Tag values. A key can contain up to 255 Unicode characters. Before verifying or using the value element, delete single-byte character (SBC) spaces before and after the value. The value can be an empty array but cannot be left blank. If no value is specified, any values are supported. Values are in an Or relationship. The system does not verify the character set of bvaluesb when searching for resources, but only verifies the length. Maximum: 255 Array Length: 0 - 20

Status code: 400

Table 4-795 Response body parameters

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

Status code: 401

Table 4-796 Response body parameters

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

Status code: 403

Table 4-797 Response body parameters

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

Status code: 404

Table 4-798 Response body parameters

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

Status code: 500

Table 4-799 Response body parameters

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

Example Requests

Querying All Resource Tag Sets of a Tenant Instance Type

```
GET https://{endpoint}/v1/resource-manager/{resource_type}/tags
```

Example Responses

Status code: 200

Succeeded.

```
{
  "tags": [ {
    "key": "key1",
    "value": [ "value1", "value2" ]
  }, {
    "key": "key2",
    "value": [ "value1", "value2" ]
  } ],
  "total_count": 1001
}
```

Status Codes

Status Code	Description
200	Succeeded.
400	Invalid parameter.
401	Unauthorized.
403	Forbidden.
404	Resources not found.
500	Internal server error.

Error Codes

See [Error Codes](#).

5 Permissions Policies and Supported Actions

5.1 Permissions Policies and Supported Actions

This chapter describes how to use IAM to implement fine-grained permissions control for your Config resources. If your account does not need individual IAM users, skip this chapter.

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

Based on authorization granularity, permissions are classified into [Roles](#) and [Policies](#).

- Roles are a type of service-based, coarse-grained authorization mechanism that defines permissions related to user responsibilities.
- Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

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

An account has all the permissions required to call all APIs, but IAM users must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user wants to query a resource recorder using an API, the user must have the permissions that allow the **rms:trackerConfig:get** action.

Supported Actions

Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Statements in a policy that allow or deny certain operations.
- APIs: REST APIs that can be called by a user who has been granted specific permissions.
- 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.
- IAM projects or 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 only take effect for IAM. For details about the differences between IAM and enterprise projects, see [What Are the Differences Between IAM and Enterprise Management?](#)

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

Resource List describes the actions and corresponding APIs for viewing resource history, listing specific resources, and other related operations.

Resource Recorder describes the actions and corresponding APIs for querying, enabling, and disabling the resource recorder.

Resource Compliance describes the actions and corresponding APIs for adding, modifying, and deleting Config rules.

Advanced Queries describes the actions and corresponding APIs for running ResourceQL queries, creating queries, and listing all queries.

Resource Aggregation describes the actions and corresponding APIs for creating, updating, and deleting resource aggregators.

Conformance Packages describes the actions and corresponding APIs for creating, listing, and deleting conformance packages.

Resource Tag the actions and corresponding APIs for listing resources, querying the number of resources, batch adding and deleting resource tags, and querying resource and project tags.

 **NOTE**

The check mark (√) indicates that an action takes effect. The cross mark (x) indicates that an action does not take effect.

5.2 Resource List

Permission	API	Action	IAM Project	Enterprise Project
Viewing change records of a resource	GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/history	rms:resources:getHistory	√	N/A

Permission	API	Action	IAM Project	Enterprise Project
Listing resource relations hips	GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/relations	rms:resources:summarize	√	N/A
Querying details about resource relations hips	GET /v1/resource-manager/domains/{domain_id}/all-resources/{resource_id}/relations	rms:resources:getRelation	√	N/A
Querying resources of a specific type	GET /v1/resource-manager/domains/{domain_id}/provider/{provider}/type/{type}/resources	rms:resources:list	√	N/A
Querying a resource	GET /v1/resource-manager/domains/{domain_id}/provider/{provider}/type/{type}/resources/{resource_id}	rms:resources:get	√	N/A
Listing all the resources recorded by the resource recorder	GET /v1/resource-manager/domains/{domain_id}/tracked-resources	rms:resources:list	√	N/A
Querying the number of resources recorded by the resource recorder	GET /v1/resource-manager/domains/{domain_id}/tracked-resources/count	rms:resources:list	√	N/A
Listing tags of resources recorded by the resource recorder	GET /v1/resource-manager/domains/{domain_id}/tracked-resources/tags	rms:resources:list	√	N/A

Permission	API	Action	IAM Project	Enterprise Project
Querying resource overview	GET /v1/resource-manager/domains/{domain_id}/tracked-resources/summary	rms:resources:list	√	N/A
Querying a specific resource recorded by the resource recorder	GET /v1/resource-manager/domains/{domain_id}/tracked-resources/{resource_id}	rms:resources:get	√	N/A
Querying a resource summary	GET /v1/resource-manager/domains/{domain_id}/all-resources/summary	rms:resources:list	√	N/A
Querying the number of resources	GET /v1/resource-manager/domains/{domain_id}/all-resources/count	rms:resources:list	√	N/A
Querying all resources	GET /v1/resource-manager/domains/{domain_id}/all-resources	rms:resources:list	√	N/A
Querying supported cloud services, resources, and regions	GET /v1/resource-manager/domains/{domain_id}/providers	Not required	N/A	N/A
Listing resource tags	GET /v1/resource-manager/domains/{domain_id}/all-resources/tags	rms:resources:list	√	N/A
Querying a resource	GET /v1/resource-manager/domains/{domain_id}/all-resources/{resource_id}	rms:resources:get	√	N/A

5.3 Resource Recorder

Permission	API	Action	IAM Project	Enterprise Project
Querying the resource recorder	GET /v1/resource-manager/domains/{domain_id}/tracker-config	rms:trackerConfig:get	√	N/A
Creating or modifying the resource recorder	PUT /v1/resource-manager/domains/{domain_id}/tracker-config	rms:trackerConfig:put	√	N/A
Deleting the resource recorder	DELETE /v1/resource-manager/domains/{domain_id}/tracker-config	rms:trackerConfig:delete	√	N/A

5.4 Resource Compliance

Permission	API	Action	Dependencies	IAM Project	Enterprise Project
Querying all built-in policies	GET /v1/resource-manager/policyDefinitions	rms:policyDefinitions:get	-	√	N/A
Querying a built-in policy	GET /v1/resource-manager/policy-definitions/{policy_definition_id}	rms:policyDefinitions:get	-	√	N/A
Modifying an assignment	PUT /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}	rms:policyAssignments:update	-	√	N/A

Permission	API	Action	Dependencies	IAM Project	Enterprise Project
Enabling an assignment	POST /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/enable	rms:policyAssignments:update	-	√	N/A
Disabling an assignment	POST /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/disable	rms:policyAssignments:update	-	√	N/A
Adding an assignment	PUT /v1/resource-manager/domains/{domain_id}/policy-assignments	rms:policyAssignments:create	-	√	N/A
Querying all assignments	GET /v1/resource-manager/domains/{domain_id}/policy-assignments	rms:policyAssignments:get	-	√	N/A
Querying an assignment	GET /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}	rms:policyAssignments:get	-	√	N/A
Deleting an assignment	DELETE /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}	rms:policyAssignments:delete	-	√	N/A
Querying the evaluation status of an assignment	GET /v1/resource-manager/domains/{domain_id}/policy-states/evaluation-state	rms:policyStates:get	-	√	N/A

Permission	API	Action	Dependencies	IAM Project	Enterprise Project
Querying the compliance result of a resource	GET /v1/resource-manager/domains/{domain_id}/resources/{resource_id}/policy-states	rms:policyStates:get	-	√	N/A
Querying all compliance results of resources under a user	GET /v1/resource-manager/domains/{domain_id}/policy-states	rms:policyStates:get	-	√	N/A
Querying all compliance results of resources evaluated by an assignment	GET /v1/resource-manager/domains/{domain_id}/policy-assignments/{policy_assignment_id}/policy-states	rms:policyStates:get	-	√	N/A
Running a resource compliance evaluation	POST /v1/resource-manager/domains/{domain_id}/policy-states/run-evaluation	rms:policyStates:runEvaluation	-	√	N/A
Updating the compliance evaluation result	GET /v1/resource-manager/domains/{domain_id}/policy-states	rms:policyStates:update	-	√	N/A

Permission	API	Action	Dependencies	IA M Project	Enterprise Project
Creating or updating an organization assignment	PUT /v1/resource-manager/organizations/{organization_id}/policy-assignments	rms:organizationPolicyAssignments:put	<ul style="list-style-type: none"> organizations:organizations:get organizations:accounts:list organizations:delegatedAdministrators:list organizations:trustedServices:enable organizations:trustedServices:list 	√	N/A
Querying organization assignments	GET /v1/resource-manager/organizations/{organization_id}/policy-assignments	rms:organizationPolicyAssignments:list	organizations:organizations:get	√	N/A
Querying a specific organization assignment	GET /v1/resource-manager/organizations/{organization_id}/policy-assignments/{organization_policy_assignment_id}	rms:organizationPolicyAssignments:get	organizations:organizations:get	√	N/A
Querying statuses of organization assignments	GET /v1/resource-manager/organizations/{organization_id}/policy-assignment-statuses	rms:organizationPolicyAssignments:list	organizations:organizations:get	√	N/A

Permission	API	Action	Dependencies	IAM Project	Enterprise Project
Querying details about rule deployment for each member account in an organization	GET /v1/resource-manager/organizations/{organization_id}/policy-assignment-detailed-status	rms:organizationPolicyAssignments:list	organizations:organizations:get	√	N/A
Deleting an organization assignment	DELETE /v1/resource-manager/organizations/{organization_id}/policy-assignments/{organization_policy_assignment_id}	rms:organizationPolicyAssignments:delete	organizations:organizations:get	√	N/A

5.5 Advanced Queries

Permissions	API	Action	IAM Project	Enterprise Project
Running an advanced query	POST /v1/resource-manager/domains/{domain_id}/run-query	rms:resources:runQuery	√	N/A
Creating a query	POST /v1/resource-manager/domains/{domain_id}/stored-queries	rms:storedQueries:create	√	N/A
Viewing a query	GET /v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}	rms:storedQueries:get	√	N/A
Listing all queries	GET /v1/resource-manager/domains/{domain_id}/stored-queries	rms:storedQueries:list	√	N/A

Permissions	API	Action	IAM Project	Enterprise Project
Updating a query	PUT /v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}	rms:storedQueries:update	√	N/A
Deleting a query	DELETE /v1/resource-manager/domains/{domain_id}/stored-queries/{query_id}	rms:storedQueries:delete	√	N/A
Listing advanced query schemas	GET /v1/resource-manager/domains/{domain_id}/schemas	rms:schemas:list	√	N/A

5.6 Resource Aggregation

Permissions	API	Action	Dependencies	IAM Project	Enterprise project
Authorizing a resource aggregator account	PUT /v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization	rms:aggregationAuthorizations:create	-	√	N/A
Deleting Authorization for a resource aggregator account	DELETE /v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization/{authorized_account_id}	rms:aggregationAuthorizations:delete	-	√	N/A
Querying authorization for a resource aggregator account	GET /v1/resource-manager/domains/{domain_id}/aggregators/aggregation-authorization	rms:aggregationAuthorizations:list	-	√	N/A

Permissions	API	Action	Dependencies	IA M Project	Enterprise project
Creating a resource aggregator	PUT /v1/resource-manager/domains/{domain_id}/aggregators	rms:aggregators:create	<ul style="list-style-type: none"> organizations:organizations:get organizations:accounts:list organizations:delegatedAdministrators:list organizations:trustedServices:enable organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A
Querying resource aggregators	GET /v1/resource-manager/domains/{domain_id}/aggregators	rms:aggregators:list	-	√	N/A
Querying details about a specific resource aggregator	GET /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}	rms:aggregators:get	-	√	N/A
Querying statuses of accounts aggregated by a specified aggregator	GET /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}/aggregator-sources-status	rms:aggregators:get	-	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise project
Deleting a resource aggregator	DELETE /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}	rms:aggregators:delete	-	√	N/A
Updating a resource aggregator	PUT /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}	rms:aggregators:update	<ul style="list-style-type: none"> organizations:organizations:get organizations:accounts:list organizations:delegatedAdministrators:list organizations:trustedServices:enable organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A
Deleting pending authorization requests	DELETE /v1/resource-manager/domains/{domain_id}/aggregators/pending-aggregation-request/{requester_account_id}	rms:aggregationRequests:delete	-	√	N/A
Querying pending aggregation requests	GET /v1/resource-manager/domains/{domain_id}/aggregators/pending-aggregation-request	rms:aggregationRequests:list	-	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise project
Querying aggregated resources	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/aggregate-discovered-resources	rms:aggregatorResources:list	<ul style="list-style-type: none"> organizations:organizations:get organizations:delegatedAdministrators:list organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A
Performing an advanced query on a specific aggregator	POST /v1/resource-manager/domains/{domain_id}/aggregators/{aggregator_id}/run-query	rms:aggregatorResources:runQuery	-	√	N/A
Querying details about a specific resource from a source account	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-resource-config	rms:aggregatorResources:get	-	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise project
Querying the number of resources from an aggregator account	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/aggregate-discovered-resource-counts	rms:aggregatorResources:list	<ul style="list-style-type: none"> organizations:organizations:get organizations:delegatedAdministrators:list organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A
Querying the compliance overview of one or more accounts in an aggregator	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-states/compliance-summary	rms:aggregatorResources:list	<ul style="list-style-type: none"> organizations:organizations:get organizations:delegatedAdministrators:list organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise project
Querying aggregated assignments	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-assignments/compliance	rms:aggregatorResources:list	<ul style="list-style-type: none"> organizations:organizations:get organizations:delegatedAdministrators:list organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A
Querying evaluation results of aggregated resources	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-states/compliance-details	rms:aggregatorResources:list	<ul style="list-style-type: none"> organizations:organizations:get organizations:delegatedAdministrators:list organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A

Permissions	API	Action	Dependencies	IA M Project	Enterprise project
Querying details about a specified aggregated assignment	POST /v1/resource-manager/domains/{domain_id}/aggregators/aggregate-data/policy-assignment/detail	rms:aggregatorResources:list	<ul style="list-style-type: none"> organizations:organizations:get organizations:delegatedAdministrators:list organizations:trustedServices:list <p>NOTE Only organization aggregators require these dependencies.</p>	√	N/A

5.7 Conformance Packages

Permissions	API	Action	Dependencies	IA M Project	Enterprise Project
Creating conformance packages	POST /v1/resource-manager/domains/{domain_id}/conformance-packs	rms:conformancePacks:create	<ul style="list-style-type: none"> rf:stack:createStack rf:stack:getStackMetadata rf:stack:listStackResources 	√	N/A
Viewing conformance packages	GET /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}	rms:conformancePacks:get	-	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise Project
Listing conformance packages	GET /v1/resource-manager/domains/{domain_id}/conformance-packs	rms:conformancePacks:list	-	√	N/A
Deleting conformance packages	DELETE /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}	rms:conformancePacks:delete	<ul style="list-style-type: none"> rf:stack:deleteStack rf:stack:getStackMetadata 	√	N/A
Updating conformance packages	PUT /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}	rms:conformancePacks:update	<ul style="list-style-type: none"> rf:stack:deployStack rf:stack:getStackMetadata rf:stack:listStackResources 	√	N/A
Listing evaluation results of all conformance packages	GET /v1/resource-manager/domains/{domain_id}/conformance-packs/compliance/summary	rms:conformancePacks:list	-	√	N/A
Listing evaluation results of a specific conformance package	GET /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}/compliance	rms:conformancePacks:get	-	√	N/A
Listing evaluation result details of a conformance package	GET /v1/resource-manager/domains/{domain_id}/conformance-packs/{conformance_pack_id}/compliance/details	rms:conformancePacks:get	-	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise Project
Listing scores of conformance packages	GET /v1/resource-manager/domains/{domain_id}/conformance-packs/scores	rms:conformancePacks:list	-	√	N/A
Creating organization conformance packages	POST /v1/resource-manager/organizations/{organization_id}/conformance-packs	rms:organizationConformancePacks:create	<ul style="list-style-type: none"> organizations:organizations:get organizations:accounts:list organizations:delegatedAdministrators:list organizations:trustedServices:enable organizations:trustedServices:list 	√	N/A
Listing organization conformance packages	GET /v1/resource-manager/organizations/{organization_id}/conformance-packs	rms:organizationConformancePacks:list	organizations:organizations:get	√	N/A
Viewing organization conformance packages	GET /v1/resource-manager/organizations/{organization_id}/conformance-packs/{conformance_pack_id}	rms:organizationConformancePacks:get	organizations:organizations:get	√	N/A
Deleting organization conformance packages	DELETE /v1/resource-manager/organizations/{organization_id}/conformance-packs/{conformance_pack_id}	rms:organizationConformancePacks:delete	organizations:organizations:get	√	N/A

Permissions	API	Action	Dependencies	IAM Project	Enterprise Project
Updating organization conformance packages	PUT /v1/resource-manager/organizations/{organization_id}/conformance-packs/{conformance_pack_id}	rms:organizationConformancePacks:update	<ul style="list-style-type: none"> organizations:organizations:get organizations:accounts:list organizations:delegatedAdministrators:list organizations:trustedServices:enable organizations:trustedServices:list 	√	N/A
Querying deployment status of organization conformance packages	GET /v1/resource-manager/organizations/{organization_id}/conformance-packs/statuses	rms:organizationConformancePacks:list	organizations:organizations:get	√	N/A
Querying deployment status details of organization conformance packages	GET /v1/resource-manager/organizations/{organization_id}/conformance-packs/detailed-statuses	rms:organizationConformancePacks:list	organizations:organizations:get	√	N/A

5.8 Resource Tag

Permissions	API	Action	Dependencies	IAM Project	Enterprise Project
Listing resources	POST /v1/resource-manager/{resource_type}/resource-instances/filter	rms:resources:listResourcesByTag	-	√	N/A
Querying the number of resources	POST /v1/resource-manager/{resource_type}/resource-instances/count	rms:resources:listResourcesByTag	-	√	N/A
Batch adding resource tags	POST /v1/resource-manager/{resource_type}/{resource_id}/tags/create	rms:resources:tagResource	-	√	N/A
Batch deleting resource tags	POST /v1/resource-manager/{resource_type}/{resource_id}/tags/delete	rms:resources:unTagResource	-	√	N/A
Querying resource tags	GET /v1/resource-manager/{resource_type}/{resource_id}/tags	rms:resources:listTagsForResource	-	√	N/A
Query project tags	GET /v1/resource-manager/{resource_type}/tags	rms:resources:listTags	-	√	N/A

6 Appendixes

6.1 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
400	RMS.0001000 1	some region is restricted to access.	Invalid region.	Specify a valid region.
400	RMS.0001000 2	resource type invalid.	Invalid value for resource type.	Specify a valid value for resource type.
400	RMS.0001000 3	service invalid.	Invalid value for service.	Specify a valid value for service.
400	RMS.0001000 4	invalid parameter.	Invalid value for parameter.	Specify a valid value for parameter.
400	RMS.0001000 7	The max length of domainId must be less than or equal to 36.	DomainId cannot be more than 36 characters.	Enter 1 to 36 characters for domainId.
400	RMS.0001000 8	domainId is illegal.	Invalid value for domainId.	Specify a valid value for domainId.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.00010009	The max length of id must be less than or equal to 512.	Id cannot be more than 512 characters.	Enter 1 to 512 characters for id.
400	RMS.00010010	id is illegal.	Invalid value for id.	Specify a valid value for id.
400	RMS.00010011	The max length of name must be less than or equal to 256.	Name cannot be more than 256 characters.	Enter 1 to 256 characters for name.
400	RMS.00010012	name is illegal.	Invalid value for name.	Specify a valid value for name.
400	RMS.00010013	The size of resource type array must be between 1 and 100.	Invalid number of resource types. Value range: 1-100	Enter 1 to 100 resource types.
400	RMS.00010014	The max length of resource type must be less than or equal to 40.	Resource type cannot be more than 40 characters.	Enter 1 to 40 characters for resource type.
400	RMS.00010015	resource type is illegal.	Invalid value for resource type.	Specify a valid value for resource type.
400	RMS.00010016	The size of region id array must be between 1 and 10.	The number of region IDs must be 1 to 10.	Enter 1 to 10 region IDs.
400	RMS.00010017	The max length of region id must be less than or equal to 36.	Region id cannot be more than 36 characters.	Enter 1 to 36 characters for region id.
400	RMS.00010018	region id is illegal.	Invalid value for region id.	Specify a valid value for region id.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.00010019	The size of enterprise project array must be between 1 and 10.	Invalid number of enterprise projects. Value range: 1-10	Enter 1 to 10 enterprise projects.
400	RMS.00010020	enterprise project is invalid.	Invalid value for enterprise project.	Specify a valid value for enterprise project.
400	RMS.00010021	The max length of project id array must be between 1 and 10.	The number of project IDs must be 1 to 10.	Enter 1 to 10 project IDs.
400	RMS.00010022	The max length of project id must be less than or equal to 36.	Project id cannot be more than 36 characters.	Enter 1 to 36 characters for project id.
400	RMS.00010023	project id is illegal.	Invalid value for project id.	Specify a valid value for project id.
400	RMS.00010024	The minimum value of limit must be greater than or equal to 1.	Limit cannot be smaller than 1.	Specify a valid value for limit.
400	RMS.00010025	The maximum value of limit must be less than or equal to 200.	Limit cannot be more than 200.	Specify a valid value for limit.
400	RMS.00010026	The minimum value of offset must be greater than or equal to 1.	Offset cannot be smaller than 1.	Specify a valid value for offset.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.0001002 7	The maximum value of offset must be less than or equal to 1000.	Offset cannot be more than 1000.	Specify a valid value for offset.
400	RMS.0001002 8	The number of tag keys must be between 1 and 5.	Invalid number of tag keys.	Enter 1 to 5 tag keys.
400	RMS.0001002 9	The length of a tag key must be less than or equal to 128.	Tag key cannot be more than 128 characters.	Enter 1 to 128 characters for tag key.
400	RMS.0001003 0	The number of values in a tag key must be less than or equal to 5.	The number of values for a tag key cannot be more than 5.	Specify a valid number of tag key values.
400	RMS.0001003 1	The length of a tag value must be less than or equal to 255.	Tag value cannot be more than 255 characters.	Enter 1 to 255 characters for tag value.
400	RMS.0001003 2	Tag value is illegal.	Invalid value for Tag value.	Specify a valid value for Tag value.
400	RMS.0001003 3	Tag key is illegal.	Invalid value for Tag key.	Specify a valid value for Tag key.
400	RMS.0001003 4	marker is illegal.	Invalid value for marker.	Specify a valid value for marker.
400	RMS.0001003 5	The length of marker must be between 4 and 400.	Invalid marker length. Value range: 4–400 characters	Enter 4 to 400 characters for marker.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.00010036	The minimum value of limit must be greater than or equal to 1.	Limit cannot be smaller than 1.	Specify a valid value for limit.
400	RMS.00010037	The maximum value of limit must be less than or equal to 200.	Limit cannot be more than 200.	Specify a valid value for limit.
400	RMS.00010038	The minimum value of offset must be greater than or equal to 1.	Offset cannot be smaller than 1.	Specify a valid value for offset.
400	RMS.00010039	The maximum value of offset must be less than or equal to 1000.	Offset cannot be more than 1000.	Specify a valid value for offset.
400	RMS.00010040	The max length of auth action must be less than or equal to 64.	Auth action cannot be more than 64 characters.	Enter 1 to 64 characters for auth action.
400	RMS.00010041	policy assignment id is illegal.	Invalid value for policy assignment id.	Specify a valid value for policy assignment id.
400	RMS.00010042	policy definition id is illegal.	Invalid value for policy definition id.	Specify a valid value for policy definition id.
400	RMS.00010043	The max length of policy assignment id must be less than or equal to 36.	Policy assignment id cannot be more than 36 characters.	Enter 1 to 36 characters for policy definition id.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.00010044	The max length of policy definition id must be less than or equal to 36.	Policy definition id cannot be more than 36 characters.	Enter 1 to 36 characters for policy definition id.
400	RMS.00010045	The max length of name must be less than or equal to 64.	Name cannot be more than 64 characters.	Enter 1 to 64 characters for name.
400	RMS.00010046	name is illegal.	Invalid value for name.	Specify a valid value for name.
400	RMS.00010047	The max length of keyword must be less than or equal to 64.	keyword cannot be more than 64 characters.	Enter 1 to 64 characters for keyword.
400	RMS.00010048	keyword is illegal.	Invalid value for keyword .	Specify a valid value for keyword.
400	RMS.00010049	The max length of compliance state must be less than or equal to 16.	Compliance state cannot be more than 16 characters.	Enter 1 to 16 characters for compliance state.
400	RMS.00010050	compliance state is illegal.	Invalid value for compliance state.	Specify a valid value for compliance state.
400	RMS.00010051	The max length of policy assignment state must be less than or equal to 16.	Policy assignment state cannot be more than 16 characters.	Enter 1 to 16 characters for policy assignment state.
400	RMS.00010052	policy assignment state is illegal.	Invalid value for policy assignment state.	Specify a valid value for policy assignment state.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.0001005 3	X-Language header is invalid.	Invalid value for X-Language header.	Specify a valid value for X-Language header.
400	RMS.0001005 4	The max length of queryld must be less than or equal to 36.	Queryld cannot be more than 36 characters.	Enter 1 to 36 characters for Queryld.
400	RMS.0001005 5	Queryld is illegal.	Invalid value for Queryld.	Specify a valid value for Queryld.
400	RMS.0001005 6	The minimum value of limit must be greater than or equal to 1.	Limit cannot be smaller than 1.	Specify a valid value for limit.
400	RMS.0001005 7	The maximum value of limit must be less than or equal to 200.	Limit cannot be more than 200.	Specify a valid value for limit.
400	RMS.0001005 8	marker is illegal.	Invalid value for marker.	Specify a valid value for marker.
400	RMS.0001005 9	The length of marker must be between 4 and 400.	Marker must contain 4 to 400 characters.	Enter 4 to 400 characters for marker.
400	RMS.0001006 0	Incorrect resource query language.	Incorrect advanced query statement.	Check whether the advanced query statement is correct.
400	RMS.0001006 1	Stored query name is illegal.	Invalid advanced query name.	Check the request parameter.
400	RMS.0001006 2	The length of stored query name must be between 1 and 64.	The advanced query name must contain 1 to 64 characters.	Enter 1 to 64 characters for the advanced query name.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.00010064	Organization_id is illegal.	Invalid value for organization_id.	Specify a valid value for organization_id.
400	RMS.00010065	Organization_id length must be between 1 and 34.	Invalid organization_id length. Value range: 1-34 characters	Enter 1 to 34 characters for organization_id.
400	RMS.00010066	Organization_policy_assignment_id is illegal.	Invalid value for organization_policy_assignment_id.	Specify a valid value for organization_policy_assignment_id.
400	RMS.00010067	Organization_policy_assignment_id length must be between 1 and 32.	Invalid organization_policy_assignment_id length. Value range: 1-32 characters	Enter 1 to 32 characters for organization_policy_assignment_id.
400	RMS.00010068	Organization_policy_assignment_name is illegal.	Invalid value for organization_policy_assignment_name.	Specify a valid value for organization_policy_assignment_name.
400	RMS.00010069	Organization_policy_assignment_name must be between 1 and 64.	Invalid organization_policy_assignment_name length. Value range: 1-64 characters	Enter 1 to 64 characters for organization_policy_assignment_name.
400	RMS.00010070	The aggregator is not found.	Aggregator not found.	Ensure that the aggregator exists.
400	RMS.00010071	Aggregator_id is illegal.	Invalid value for aggregator_id.	Specify a valid value for aggregator_id.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.0001007 2	Aggregator_id length must be between 1 and 32.	Invalid aggregator_id length. Value range: 1-32 characters	Enter 1 to 36 characters for aggregator_id.
400	RMS.0001007 3	Authorized_account_id is illegal.	Invalid value for authorized_account_id.	Specify a valid value for authorized_account_id.
400	RMS.0001007 4	Authorized_account_id length must be between 1 and 36.	Invalid authorized_account_id length. Value range: 1-36 characters	Enter 1 to 36 characters for authorized_account_id.
400	RMS.0001007 5	Requester_account_id is illegal.	Invalid value for requester_account_id.	Specify a valid value for requester_account_id.
400	RMS.0001007 6	Requester_account_id length must be between 1 and 36.	Invalid requester_account_id length. Value range: 1-36 characters	Enter 1 to 36 characters for requester_account_id.
400	RMS.0001007 7	Aggregator_name is illegal.	Invalid value for aggregator_name.	Specify a valid value for aggregator_name.
400	RMS.0001007 8	Aggregator_name length must be between 1 and 64.	Invalid aggregator_name length. Value range: 1-64 characters	Enter 1 to 64 characters for aggregator_name.
400	RMS.0001008 0	Aggregator_update_status length must be between 1 and 16.	Invalid aggregator_update_status length. Value range: 1-16 characters	Enter 1 to 36 characters for aggregator_update_status.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.0001008 1	The maximum value of limit is 2000.	Limit cannot be more than 2000.	Specify a valid value for limit.
400	RMS.0001008 2	The number of aggregators exceed quota.	The number of aggregators exceed quota. The quota of organization aggregator is 1, account aggregator is 30.	Delete useless aggregators.
400	RMS.0001008 3	Conformance_pack_name is illegal.	Invalid value for conformance_pack_name.	Specify a valid value for conformance_pack_name.
400	RMS.0001008 4	Conformance_pack_name length must be between 1 and 64.	Invalid conformance_pack_name length. Value range: 1-64 characters	Enter 1 to 64 characters for conformance_pack_name.
400	RMS.0001008 5	Conformance_pack_id is illegal.	Invalid value for conformance_pack_id.	Specify a valid value for conformance_pack_id.
400	RMS.0001008 6	Conformance_pack_id length must be between 1 and 36.	Invalid conformance_pack_id length. Value range: 1-36 characters	Enter 1 to 36 characters for conformance_pack_id.
400	RMS.0001008 7	Template_key is illegal.	Invalid value for template_key.	Specify a valid value for template_key.
400	RMS.0001008 8	Template_key length must be between 1 and 128.	Invalid template_key length. Value range: 1-128 characters	Enter 1 to 128 characters for template_key.

Status Code	Error Codes	Error Message	Description	Solution
400	RMS.00010089	Template_id is illegal.	Invalid value for template_id.	Specify a valid value for template_id.
400	RMS.00010090	Template_id length must be between 1 and 36.	Invalid template_id length. Value range: 1-36 characters	Enter 1 to 36 characters for template_id.
400	RMS.00010091	The tracker-config is not enabled.	The tracker-config is not enabled.	Enable the tracker-config.
400	RMS.00010092	The number of accounts operated by the aggregator exceeds the limit.	The aggregator operation has exceeded the limit. The maximum number of accounts that can be added or removed from the account aggregator within seven days is 1000	Please check if the maximum number of accounts added or removed within seven days exceeds 1000.
400	RMS.00010093	Exceeded limit on organization aggregator operations.	The aggregator operation has exceeded the limit. the maximum number of new additions to the organization aggregator within one day is 1.	Please check the number of organization aggregators created within a day.
401	RMS.00010079	Incorrect IAM authentication information: Authorization header is missing.	Incorrect IAM authentication information: Authorization header is missing.	Add authentication header.

Status Code	Error Codes	Error Message	Description	Solution
403	RMS.00010006	access denied.	Access denied.	Check whether you have the required permissions.
404	RMS.00010005	resource not found.	Resource not found.	Ensure that the resource exists.
429	RMS.00010063	ResourceQL is busy.	Advanced queries are busy.	Try again later.
500	RMS.INTERNAL.ERROR	Internal server error occurred.	Service internal error occurred.	Try again later or contact customer service.

6.2 Supported Services and Resource Types

You can obtain the cloud services (**provider**) and resource types (**type**) supported by Config in either of the following ways:

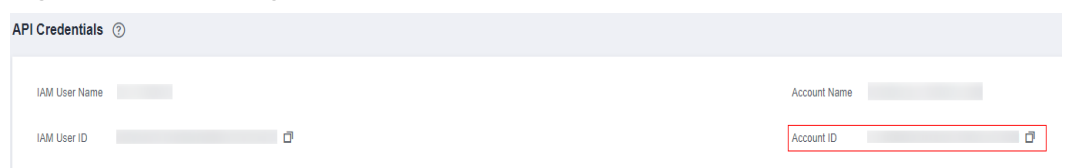
- Calling APIs
You can call the API for [listing cloud services](#) to query cloud services, resources, and regions supported by Config. The **provider** field indicates the cloud service name, and the **name** field in **resource_types** indicates the resource type name.
- Management console
For details about the services and resources supported by Config, see [Supported Services and Regions](#). You can view related information in the **Resource Type** column of the list.

6.3 Obtaining an Account ID

An account ID is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

1. Sign in to the management console.
2. Hover over the username and select **My Credentials** from the drop-down list. View the account ID on the **API Credentials** page.

Figure 6-1 Obtaining an account ID



6.4 Status Codes

- Normal

Returned Value	Description
200 OK	The results of GET and PUT operations are returned as expected.
201 Created	The results of the POST operation are returned as expected.
202 Accepted	The request has been accepted for processing.
204 No Content	The results of the DELETE operation are returned as expected.

- Abnormal

Returned Value	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and password to access the requested page.
403 Forbidden	You are forbidden to access the requested page.
404 Not Found	The server cannot find the requested page.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server cannot be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication so that the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of a service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.

Returned Value	Description
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request. The service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

6.5 Obtaining a Project ID

Scenarios

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

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

Obtain the Project ID by Calling an API

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

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

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

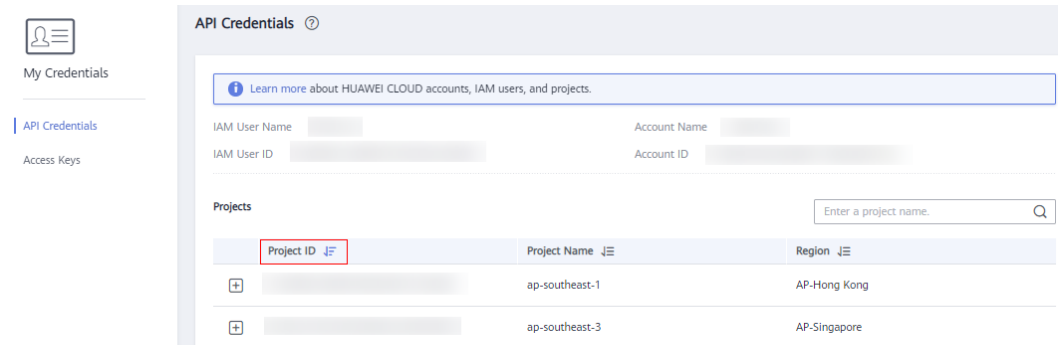
```
{
  "projects": [
    {
      "domain_id": "65ewtrgaggshhk1223245sghjlse684b",
      "is_domain": false,
      "parent_id": "65ewtrgaggshhk1223245sghjlse684b",
      "name": "project_name",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects/a4adasfjljaaakla12334jklga9sasfg"
      },
      "id": "a4adasfjljaaakla12334jklga9sasfg",
      "enabled": true
    }
  ],
  "links": {
    "next": null,
    "previous": null,
    "self": "https://www.example.com/v3/projects"
  }
}
```

Obtain a Project ID from the Console

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

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

Figure 6-2 Viewing the project ID



7 Change History

Released On	Description
2023-12-20	<p>This issue is the ninth official release, which incorporates the following changes:</p> <ul style="list-style-type: none">• Added the API for Listing all the resources recorded by the resource recorder.• Added the API for querying the number of resources recorded by the resource recorder• Added the API for Listing the tags of the resources recorded by the resource recorder• Added the API for querying resource overview.• Added the API for querying a specific resource recorded by the resource recorder.• Added supported actions of APIs related to the resource recorder to the "Resource List" section.
2023-10-20	<p>This issue is the eighth official release, which incorporates the following changes:</p> <ul style="list-style-type: none">• Added organization conformance package APIs to in the "Conformance Packages" section.• Added supported actions of organization conformance package APIs to the "Conformance Packages" section.• Updated the "API Overview" chapter.
2023-08-30	<p>This issue is the seventh official release, which incorporates the following changes:</p> <ul style="list-style-type: none">• Added organization conformance package APIs.• Added the "Conformance Packages" section to the "Permissions and Supported Actions" chapter.• Updated the "API Overview" chapter.• Updated the "Error Codes" section.

Released On	Description
2023-06-07	<p>This issue is the sixth official release, which incorporates the following change:</p> <p>Changed the service name from Resource Management Service (RMS) to Config.</p>
2023-02-17	<p>This issue is the fifth official release, which incorporates the following changes:</p> <ul style="list-style-type: none"> ● Added the API for querying a specific resource. ● Added the API for listing resource tags. ● Added the API for querying the number of resources. ● Added the API for querying resource overview. ● Added the API for querying details about resource relationships. ● Added the API for creating or updating organization rules. ● Added the API for querying organization rules. ● Added the API for querying rules of a specified organization. ● Added the API for deleting organization rules. ● Added the API for querying the deployment statuses of organization rules. ● Added the API for querying the rule deployment statuses of every organization member. ● Added APIs of resource aggregators. ● Added the "Resource Aggregation" section to the "Permissions and Supported Actions" chapter.
2022-12-07	<p>This issue is the fourth official release, which incorporates the following change:</p> <ul style="list-style-type: none"> ● Added the API for updating conformance evaluation results.
2021-04-06	<p>This issue is the third official release, which incorporates the following changes:</p> <ul style="list-style-type: none"> ● Added APIs of advanced queries. ● Added the "Advanced Queries" sections to the "Permissions and Supported Actions" chapter.
2020-12-25	<p>This issue is the second official release, which incorporates the following changes:</p> <ul style="list-style-type: none"> ● Added error code RMS.00010053. ● Added the API for listing all resources. ● Added the API for querying available regions.
2020-11-30	<p>This issue is the first official release.</p>