

Resource Access Manager

API Reference

Issue 01
Date 2024-03-15



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 Overview.....	1
1.2 API Calling.....	1
1.3 Endpoints.....	1
1.4 Concepts.....	1
2 API Overview.....	4
3 Calling APIs.....	5
3.1 Making an API Request.....	5
3.2 Authentication.....	7
3.3 Response.....	8
4 APIs.....	10
4.1 RAM Managed Permissions.....	10
4.1.1 Getting a List of RAM Managed Permissions.....	10
4.1.2 Getting Details About RAM Managed Permissions.....	14
4.1.3 Getting all versions of permissions.....	17
4.2 Resource Shares.....	20
4.2.1 Creating a Resource Share.....	20
4.2.2 Searching for a Resource Share.....	25
4.2.3 Updating a Resource Share.....	29
4.2.4 Deleting a Resource Share.....	33
4.3 Associated Principals and Resources.....	34
4.3.1 Associating Principals and Resources.....	34
4.3.2 Disassociating Principals and Resources.....	37
4.3.3 Searching for Associated Principals and Resources.....	40
4.4 Associated RAM Managed Permissions.....	44
4.4.1 Associating or Replacing RAM Managed Permissions.....	44
4.4.2 Disassociating RAM Managed Permissions.....	46
4.4.3 Getting Associated RAM Managed Permissions.....	48
4.5 Tag Management.....	51
4.5.1 Querying Tags Used by Resource Shares.....	51
4.5.2 Querying Resource Shares by Tag.....	54
4.5.3 Querying Number of Resource Shares by Tag.....	58

4.5.4 Adding Tags to Resource Shares.....	61
4.5.5 Deleting Tags from Resource Shares.....	63
4.6 Shared Resources.....	65
4.6.1 Searching for Shared Resources.....	65
4.7 Principals.....	69
4.7.1 Searching for Principals.....	69
4.8 Resource Sharing Invitations.....	72
4.8.1 Accepting a Resource Sharing Invitation.....	72
4.8.2 Rejecting a Resource Sharing Invitation.....	75
4.8.3 Searching for a Resource Sharing Invitation.....	77
4.9 Sharing with Organizations.....	81
4.9.1 Checking Whether Sharing with Organizations Is Enabled.....	81
4.9.2 Enabling Sharing with Organizations.....	83
4.9.3 Disabling Sharing with Organizations.....	84
4.10 Other operations.....	85
4.10.1 Querying Resource Sharing Quotas.....	85
4.10.2 Querying Resource Type.....	88
5 Permissions and Supported Actions.....	91
5.1 Introduction.....	91
5.2 Actions.....	92
6 Appendixes.....	96
6.1 Status Codes.....	96
6.2 Error Codes.....	99
6.3 Obtaining Information About Account, IAM User, Group, Project, Region, and Agency.....	108
7 Change History.....	110

1 Before You Start

1.1 Overview

Welcome to use the Resource Access Manager (RAM) service. RAM is a service that helps you securely share your cloud resources across accounts. You can use RAM to share the resources that you create in one of your accounts with other accounts. With RAM, you can also centrally manage shared resources.

This document describes RAM application programming interfaces (APIs), including the API descriptions, parameters, and examples. You can use the APIs to perform RAM-related operations, such as creating, deleting, modifying, and querying. For details about all supported operations, see [API Overview](#).

If you plan to use an API to access RAM, ensure that you are familiar with RAM concepts described in [Service Overview](#).

1.2 API Calling

RAM supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For more details, see [Calling APIs](#).

1.3 Endpoints

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

1.4 Concepts

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

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

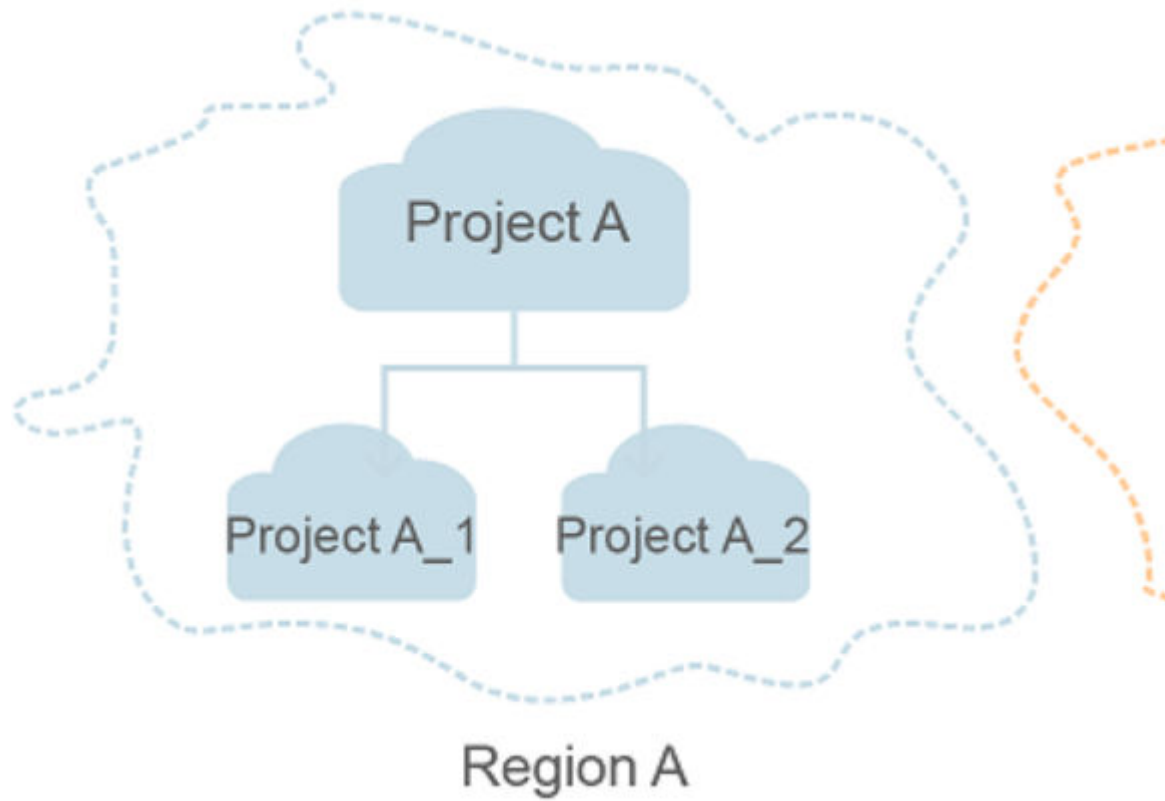
Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region.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

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

An availability zone (AZ) comprises one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Compute, 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. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolating model



2 API Overview

Table 2-1 RAM APIs

Type	Description
RAM managed permissions	List RAM managed permissions, get their details for the specified resource type, and obtain all permission versions.
Resource shares	Create, list, update, or delete resource shares.
Associated principals and resources	Associate, disassociate, or list principals and resources for a resource share.
Associated RAM managed permissions	Associate or replace, disassociate, or list RAM managed permissions for a resource share.
Tag management	Add, delete, update, and query tags attached to resource shares.
Shared resources	List shared resources in a resource share.
Principals	List principals in a resource share.
Resource sharing invitations	Accept, reject, or list resource sharing invitations.
Sharing with Organizations	Check whether sharing with Organizations is enabled or enable/disable sharing with Organizations.
Other operations	Query resource sharing quotas and search for resource types of cloud services.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request. The API in [Getting a List of RAM Managed Permissions](#) is taken as an example to demonstrate how to call an API.

Request URI

A request URI is in the following format:

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

Table 3-1 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 RAM is ram.myhuaweicloud.com .
resource-path	Access path of an API for performing the specified operation. You can obtain the path from the URI of an API. For example, the resource-path of the API used to list RAM managed permissions for the specified resource type is /v1/permissions .
query-string	An optional query parameter. Ensure that a question mark (?) is included before each query parameter that is in the format of Parameter name = Parameter value . For example, ?limit=10 indicates that a maximum of 10 data records will be queried.

For example, to list RAM managed permissions for the specified resource type, obtain the endpoint of RAM (**ram.myhuaweicloud.com**) and the resource-path

(/v1/permissions) in [Getting a List of RAM Managed Permissions](#). Then, construct the URI as follows:

Request Methods

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

Table 3-2 REST request methods supported by RAM

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.

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. [Table 3-3](#) describes common request headers.

The following headers need to be added to the request:

Table 3-3 Common request headers

Header	Description	Mandatory
Authorization	Signature information in the request. For details about AK/SK authentication, see AK/SK Authentication . Type: string Default value: none	Yes
Content-Type	Content type, for example, application/json. Type: string Default value: none	Yes
X-Sdk-Date	Date and time when the request was sent, for example, 20221107T020014Z . Type: string Default value: none	Yes

Header	Description	Mandatory
Host	Host address, for example, ram.myhuaweicloud.com. Type: string Default value: none	Yes

 NOTE

APIs support AK/SK authentication, which uses SDKs to sign a request. During the signature, the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) headers are automatically added in the request. For details about AK/SK authentication, see [AK/SK Authentication](#).

For the API in [Getting a List of RAM Managed Permissions](#), the request is as follows:

```
GET https://ram.myhuaweicloud.com/v1/permissions
content-type: application/json
X-Sdk-Date: 20230330T021902Z
host: ram.myhuaweicloud.com
Authorization: SDK-HMAC-SHA256 Access=xxxxxxxxxxxxxxxxxxxxx, SignedHeaders=content-type;host;x-sdk-date, Signature=xxxxxxxxxxxxxxxxxxxxx
```

(Optional) Request Body

The body of a request is often sent in a structured format (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.

Initiating a Request

You can send the request to call an API through [curl](#), [Postman](#), or coding.

3.2 Authentication

AK/SK authentication is used for calling APIs. Specifically, requests are encrypted using the access key (AK) and secret key (SK) to provide higher security.

AK/SK Authentication

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 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 use the signing SDK to sign requests. For details about how to sign requests and use the signing SDK, see [API Request Signing Guide](#).

NOTICE

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

3.3 Response

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

Status Code

A status code is a group of digits ranging from 2xx (indicating successes) to 4xx or 5xx (indicating errors). It indicates the status of a response. For more information, see [Status Codes](#).

Response Header

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

[Table 3-4](#) describes common response headers.

Table 3-4 Common response headers

Header	Description
Content-Type	Type of the resource content. Type: string Default value: none
Connection	Whether the connection to the server is a long connection or a short connection. Type: string Valid values: keep-alive close Default value: none
Date	Date when the RAM service responded to the request. Type: string Default value: none
X-Request-Id	Uniquely identifies the request. The value is generated by the RAM service and can be used for troubleshooting. Type: string Default value: none

Response Body

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

For the API in [Getting a List of RAM Managed Permissions](#), the following message body is returned:

```
{
  "permissions": [
    {
      "id": "7d04b6d4-00d5-41bd-b13c-22abd1ea249b",
      "name": "test-dns-zones",
      "resource_type": "dns:zone",
      "is_resource_type_default": true,
      "created_at": "2022-08-10T07:18:47Z",
      "updated_at": "2022-10-21T15:22:15Z"
    },
    {
      "id": "8c378992-2450-4b2e-9c94-58fe01c74397",
      "name": "test-dns-resolverRules",
      "resource_type": "dns:resolverRule",
      "is_resource_type_default": true,
      "created_at": "2022-08-10T07:50:02Z",
      "updated_at": "2022-10-25T17:27:01Z"
    },
    .....
  ]
}
```

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 resource type vpc:subnet has no permission",
  "error_code": "RAM.1009"
}
```

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

4 APIs

4.1 RAM Managed Permissions

4.1.1 Getting a List of RAM Managed Permissions

Function

This API is used to get a list of RAM managed permissions for the specified resource type.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/permissions

Table 4-1 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64

Parameter	Mandatory	Type	Description
resource_type	No	String	Name of the resource type. Minimum: 0 Maximum: 64
permission_type	No	String	Type of the permission. RAM_MANAGED indicates RAM managed permissions, CUSTOMER_MANAGED indicates permissions created by tenants, and ALL indicates both permission types. Default: ALL Minimum: 0 Maximum: 64

Request Parameters

Table 4-2 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 4-3 Response body parameters

Parameter	Type	Description
permissions	Array of PermissionSummary objects	List of details about RAM managed permissions. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-4 PermissionSummary

Parameter	Type	Description
id	String	Permission ID. Minimum: 0 Maximum: 36
name	String	Permission name. Minimum: 1 Maximum: 128
resource_type	String	Resource type to which the permission applies. Minimum: 0 Maximum: 100
is_resource_type_default	Boolean	Whether the permission is the default permission for the resource type.
created_at	String	Time when the permission was created.
updated_at	String	Time when the permission was last updated.
permission_urn	String	URN for the permission. Minimum: 1 Maximum: 1500
permission_type	String	Permission type, either RAM managed or custom permission. Minimum: 0 Maximum: 64
default_version	Boolean	Whether the current version is the default version.
version	Integer	Version of the permission. Minimum: 0 Maximum: 1000000
status	String	Status of the permission. Minimum: 0 Maximum: 64

Table 4-5 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Getting a list of resource sharing permissions for the specified resource type

```
GET https://{hostname}/v1/permissions
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "permissions": [ {
    "id": "string",
    "name": "string",
    "resource_type": "string",
    "is_resource_type_default": true,
    "created_at": "2022-09-19T02:02:18.155Z",
    "updated_at": "2022-09-19T02:02:18.155Z",
    "permission_urn": "string",
    "permission_type": "string",
    "default_version": false,
    "version": 1,
    "status": "string"
  } ],
  "page_info": {
    "previous_marker": "string",
    "next_marker": "string",
    "current_count": 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.1.2 Getting Details About RAM Managed Permissions

Function

This API is used to get the details of RAM managed permissions of the specified version for the specified resource type. If the permission version is not specified, the information about the default permission version is returned.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/permissions/{permission_id}

Table 4-6 Path Parameters

Parameter	Mandatory	Type	Description
permission_id	Yes	String	ID of the RAM managed permission. Minimum: 0 Maximum: 36

Table 4-7 Query Parameters

Parameter	Mandatory	Type	Description
permission_version	No	Integer	Version of the permission. Minimum: 0 Maximum: 1000000

Request Parameters

Table 4-8 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 4-9 Response body parameters

Parameter	Type	Description
permission	Permission object	Describes information about RAM permissions.

Table 4-10 Permission

Parameter	Type	Description
id	String	Permission ID. Minimum: 0 Maximum: 36
name	String	Permission name. Minimum: 1 Maximum: 128
resource_type	String	Resource type. Minimum: 0 Maximum: 100
content	String	Impact and actions allowed by the permission. Minimum: 1 Maximum: 65535
is_resource_type_default	Boolean	Whether the permission is the default permission for the resource type.
created_at	String	Time when the permission was created.

Parameter	Type	Description
updated_at	String	Time when the permission was last updated.
permission_urn	String	URN for the permission. Minimum: 1 Maximum: 1500
permission_type	String	Type of the permission. Minimum: 0 Maximum: 64
default_version	Boolean	Whether the current version is the default version.
version	Integer	Version of the permission. Minimum: 0 Maximum: 1000000
status	String	Status of the permission. Minimum: 0 Maximum: 64

Example Requests

Getting details of RAM managed permissions of the specified version for the specified resource type (default permission version used if not specified)

```
GET https://{hostname}/v1/permissions/{permission_urn}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "permission" : {
    "id" : "string",
    "name" : "string",
    "resource_type" : "string",
    "content" : "string",
    "is_resource_type_default" : true,
    "created_at" : "2022-08-22T11:40:31.871Z",
    "updated_at" : "2022-08-22T11:40:31.871Z",
    "permission_urn" : "string",
    "permission_type" : "string",
    "default_version" : false,
    "version" : 1,
    "status" : "string"
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.1.3 Getting all versions of permissions

Function

This API is used to get all versions of permissions.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/permissions/{permission_id}/versions

Table 4-11 Path Parameters

Parameter	Mandatory	Type	Description
permission_id	Yes	String	ID of the RAM managed permission. Minimum: 0 Maximum: 36

Table 4-12 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50

Parameter	Mandatory	Type	Description
marker	No	String	Page marker. Minimum: 1 Maximum: 64

Request Parameters

Table 4-13 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 4-14 Response body parameters

Parameter	Type	Description
permissions	Array of PermissionSummary objects	List of details about RAM managed permissions. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-15 PermissionSummary

Parameter	Type	Description
id	String	Permission ID. Minimum: 0 Maximum: 36
name	String	Permission name. Minimum: 1 Maximum: 128

Parameter	Type	Description
resource_type	String	Resource type to which the permission applies. Minimum: 0 Maximum: 100
is_resource_type_default	Boolean	Whether the permission is the default permission for the resource type.
created_at	String	Time when the permission was created.
updated_at	String	Time when the permission was last updated.
permission_urn	String	URN for the permission. Minimum: 1 Maximum: 1500
permission_type	String	Permission type, either RAM managed or custom permission. Minimum: 0 Maximum: 64
default_version	Boolean	Whether the current version is the default version.
version	Integer	Version of the permission. Minimum: 0 Maximum: 1000000
status	String	Status of the permission. Minimum: 0 Maximum: 64

Table 4-16 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

None

Example Responses

Status code: 200

Request succeeded.

```
{
  "permissions": [ {
    "id": "string",
    "name": "string",
    "resource_type": "string",
    "is_resource_type_default": true,
    "created_at": "2022-09-19T02:02:18.155Z",
    "updated_at": "2022-09-19T02:02:18.155Z",
    "permission_urn": "string",
    "permission_type": "string",
    "default_version": false,
    "version": 1,
    "status": "string"
  } ],
  "page_info": {
    "previous_marker": "string",
    "next_marker": "string",
    "current_count": 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.2 Resource Shares

4.2.1 Creating a Resource Share

Function

This API is used to create a resource share. You can specify the list of resources to be shared, the list of principals in resource sharing, and the list of permissions granted to the principals.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares

Request Parameters

Table 4-17 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 4-18 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the resource share. Minimum: 1 Maximum: 64
description	No	String	Description of the resource share. Minimum: 0 Maximum: 256
allow_external_principals	No	Boolean	Whether resources can be shared with any accounts outside the organization. Default: true

Parameter	Mandatory	Type	Description
permission_ids	No	Array of strings	List of RAM permissions associated with the resource share. A resource type can be associated with only one RAM permission. If you do not specify a permission ID, RAM automatically associates the default permission for each resource type. Minimum: 0 Maximum: 36 Array Length: 0 - 20
principals	No	Array of strings	List of one or more principals associated with the resource share. Minimum: 0 Maximum: 1500 Array Length: 0 - 20
resource_urns	No	Array of strings	List of URNs of one or more resources associated with the resource share. Minimum: 0 Maximum: 1500 Array Length: 0 - 20
tags	No	Array of Tag objects	List of tags attached to the resource share. Array Length: 0 - 20

Table 4-19 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	Yes	String	Tag value. You can set the tag value to an empty string but not to null. Minimum: 0 Maximum: 255

Response Parameters

Status code: 201

Table 4-20 Response body parameters

Parameter	Type	Description
resource_share	ResourceShare object	Describes a resource share in RAM.

Table 4-21 ResourceShare

Parameter	Type	Description
id	String	ID of the resource share. Minimum: 0 Maximum: 36
name	String	Name of the resource share. Minimum: 0 Maximum: 128
description	String	Description of the resource share. Minimum: 1 Maximum: 1024
allow_external_principals	Boolean	Whether resources can be shared with any accounts outside the organization.
owning_account_id	String	ID of the resource owner in a resource share. Minimum: 0 Maximum: 36
status	String	Status of the resource share. Minimum: 0 Maximum: 36
tags	Array of Tag objects	List of tags attached to the resource share. Array Length: 0 - 100
created_at	String	Time when the resource share was created.
updated_at	String	Time when the resource share was last updated.

Table 4-22 Tag

Parameter	Type	Description
key	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	String	Tag value. You can set the tag value to an empty string but not to null. Minimum: 0 Maximum: 255

Example Requests

Creating a resource share, specifying the resources and principals to be associated, granting permissions to the principals, and adding tags to the resource share

```
POST https://{hostname}/v1/resource-shares
```

```
{
  "name" : "string",
  "description" : "string",
  "permission_ids" : [ "string" ],
  "principals" : [ "string" ],
  "resource_urns" : [ "string" ],
  "tags" : [ {
    "key" : "string",
    "value" : "string"
  } ]
}
```

Example Responses

Status code: 201

Request succeeded.

```
{
  "resource_share" : {
    "id" : "string",
    "name" : "string",
    "description" : "string",
    "allow_external_principals" : true,
    "owning_account_id" : "string",
    "status" : "string",
    "tags" : [ {
      "key" : "string",
      "value" : "string"
    } ],
    "created_at" : "2022-09-27T03:14:08.883Z",
    "updated_at" : "2022-09-27T03:14:08.883Z"
  }
}
```

Status Codes

Status Code	Description
201	Request succeeded.

Error Codes

See [Error Codes](#).

4.2.2 Searching for a Resource Share

Function

This API is used to search for the details about the resource share that you have created or that shared with you.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/search

Request Parameters

Table 4-23 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 4-24 Request body parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
name	No	String	Name of the resource share. Minimum: 0 Maximum: 64
marker	No	String	Page marker. Minimum: 1 Maximum: 64
permission_id	No	String	Permission ID. Minimum: 0 Maximum: 36
resource_owner	Yes	String	Whether the resource share is created by you (self) or shared with you (other-accounts).
resource_share_ids	No	Array of strings	List of resource share IDs. Minimum: 0 Maximum: 36 Array Length: 1 - 100
resource_share_status	No	String	Status of the resource share. Minimum: 0 Maximum: 36
tag_filters	No	Array of TagFilter objects	Tags attached to the resource share. Array Length: 0 - 20

Table 4-25 TagFilter

Parameter	Mandatory	Type	Description
key	Yes	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128

Parameter	Mandatory	Type	Description
values	No	Array of strings	List of values for the tag key. Minimum: 0 Maximum: 255 Array Length: 0 - 20

Response Parameters

Status code: 200

Table 4-26 Response body parameters

Parameter	Type	Description
resource_shares	Array of ResourceShare objects	List of details about resource shares. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-27 ResourceShare

Parameter	Type	Description
id	String	ID of the resource share. Minimum: 0 Maximum: 36
name	String	Name of the resource share. Minimum: 0 Maximum: 128
description	String	Description of the resource share. Minimum: 1 Maximum: 1024
allow_external_principals	Boolean	Whether resources can be shared with any accounts outside the organization.
owning_account_id	String	ID of the resource owner in a resource share. Minimum: 0 Maximum: 36

Parameter	Type	Description
status	String	Status of the resource share. Minimum: 0 Maximum: 36
tags	Array of Tag objects	List of tags attached to the resource share. Array Length: 0 - 100
created_at	String	Time when the resource share was created.
updated_at	String	Time when the resource share was last updated.

Table 4-28 Tag

Parameter	Type	Description
key	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	String	Tag value. You can set the tag value to an empty string but not to null. Minimum: 0 Maximum: 255

Table 4-29 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Searching for details about a resource share

POST https://{hostname}/v1/resource-shares/search

```
{
  "limit" : 2000,
  "name" : "string",
  "marker" : "string",
  "permission_id" : "string",
  "resource_owner" : "self",
  "resource_share_ids" : [ "string" ],
  "resource_share_status" : "string",
  "tag_filters" : [ {
    "key" : "string",
    "values" : [ "string" ]
  } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_shares" : [ {
    "id" : "string",
    "name" : "string",
    "description" : "string",
    "allow_external_principals" : true,
    "owning_account_id" : "string",
    "status" : "string",
    "tags" : [ {
      "key" : "string",
      "value" : "string"
    } ],
    "created_at" : "2023-01-06T08:24:55.638Z",
    "updated_at" : "2023-01-06T08:24:55.638Z"
  } ],
  "page_info" : {
    "previous_marker" : "string",
    "next_marker" : "string",
    "current_count" : 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.2.3 Updating a Resource Share

Function

This API is used to modify the properties of a resource share.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

PUT /v1/resource-shares/{resource_share_id}

Table 4-30 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-31 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 4-32 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the resource share. Minimum: 1 Maximum: 64
description	No	String	Description of the resource share. Minimum: 0 Maximum: 256
allow_external_principals	No	Boolean	Whether resources can be shared with any accounts outside the organization.

Response Parameters

Status code: 200

Table 4-33 Response body parameters

Parameter	Type	Description
resource_share	ResourceShare object	Describes a resource share in RAM.

Table 4-34 ResourceShare

Parameter	Type	Description
id	String	ID of the resource share. Minimum: 0 Maximum: 36
name	String	Name of the resource share. Minimum: 0 Maximum: 128
description	String	Description of the resource share. Minimum: 1 Maximum: 1024
allow_external_principals	Boolean	Whether resources can be shared with any accounts outside the organization.
owning_account_id	String	ID of the resource owner in a resource share. Minimum: 0 Maximum: 36
status	String	Status of the resource share. Minimum: 0 Maximum: 36
tags	Array of Tag objects	List of tags attached to the resource share. Array Length: 0 - 100
created_at	String	Time when the resource share was created.
updated_at	String	Time when the resource share was last updated.

Table 4-35 Tag

Parameter	Type	Description
key	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	String	Tag value. You can set the tag value to an empty string but not to null. Minimum: 0 Maximum: 255

Example Requests

Modifying the name or description of a resource share

```
PUT https://{hostname}/v1/resource-shares/{resource_share_id}
{
  "name" : "string",
  "description" : "string"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_share" : {
    "id" : "string",
    "name" : "string",
    "description" : "string",
    "allow_external_principals" : true,
    "owning_account_id" : "string",
    "status" : "string",
    "tags" : [ {
      "key" : "string",
      "value" : "string"
    } ],
    "created_at" : "2023-01-30T11:29:17.578Z",
    "updated_at" : "2023-01-30T11:29:17.578Z"
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.2.4 Deleting a Resource Share

Function

This API is used to delete a specified resource share. This operation does not delete the resources but stops sharing the resources with other accounts.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

DELETE /v1/resource-shares/{resource_share_id}

Table 4-36 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-37 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

Deleting a specified resource share

```
DELETE https://{hostname}/v1/resource-shares/{resource_share_id}
```

Example Responses

None

Status Codes

Status Code	Description
204	Request succeeded.

Error Codes

See [Error Codes](#).

4.3 Associated Principals and Resources

4.3.1 Associating Principals and Resources

Function

This API is used to associate a list of specified principals or resources with a resource share. If a resource is associated, the principals that have the permission to access the resource share will gain the permission to access that resource in the resource share. If a principal is associated, the principal will have the permission to access the shared resources in the resource share.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/{resource_share_id}/associate

Table 4-38 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-39 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 4-40 Request body parameters

Parameter	Mandatory	Type	Description
principals	No	Array of strings	List of one or more principals associated with the resource share. Minimum: 0 Maximum: 1500 Array Length: 0 - 20
resource_urns	No	Array of strings	List of URNs of one or more resources associated with the resource share. Minimum: 0 Maximum: 1500 Array Length: 0 - 20

Response Parameters

Status code: 200

Table 4-41 Response body parameters

Parameter	Type	Description
resource_share_associations	Array of ResourceShareAssociation objects	Details about the principals or resources associated with a resource share. Array Length: 0 - 2000

Table 4-42 ResourceShareAssociation

Parameter	Type	Description
associated_entity	String	Associated entity. It can be the resource URN, account ID, URN of the root OU, or URN of another OU. Minimum: 0 Maximum: 1500
association_type	String	Entity type in the association. Minimum: 0 Maximum: 36
created_at	String	Time when the association was created.
updated_at	String	Time when the association was last updated.
external	Boolean	Whether the principle is in the same organization as the resource owner.
resource_share_id	String	ID of the resource share. Minimum: 0 Maximum: 36
resource_share_name	String	Name of the resource share. Minimum: 0 Maximum: 64
status	String	Status of the association. Minimum: 0 Maximum: 36
status_message	String	Description of the status to the association. Minimum: 0 Maximum: 65535

Example Requests

Associating a list of specified principals or resources with a resource share

```
POST https://{hostname}/v1/resource-shares/{resource_share_id}/associate
```

```
{  
  "principals": [ "string" ],  
  "resource_urns": [ "string" ]  
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_share_associations": [ {
    "associated_entity": "string",
    "association_type": "string",
    "created_at": "2022-08-22T11:50:25.639Z",
    "updated_at": "2022-08-22T11:50:25.639Z",
    "external": true,
    "resource_share_id": "string",
    "resource_share_name": "string",
    "status": "string",
    "status_message": "string"
  } ]
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.3.2 Disassociating Principals and Resources

Function

This API is used to disassociate specified principals or resources from a specified resource share.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/{resource_share_id}/disassociate

Table 4-43 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-44 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 4-45 Request body parameters

Parameter	Mandatory	Type	Description
principals	No	Array of strings	List of one or more principals associated with the resource share. Minimum: 0 Maximum: 1500 Array Length: 0 - 20
resource_urns	No	Array of strings	List of URNs of one or more resources associated with the resource share. Minimum: 0 Maximum: 1500 Array Length: 0 - 20

Response Parameters

Status code: 200

Table 4-46 Response body parameters

Parameter	Type	Description
resource_share_associations	Array of ResourceShareAssociation objects	Details about the principals or resources associated with a resource share. Array Length: 0 - 2000

Table 4-47 ResourceShareAssociation

Parameter	Type	Description
associated_entity	String	Associated entity. It can be the resource URN, account ID, URN of the root OU, or URN of another OU. Minimum: 0 Maximum: 1500
association_type	String	Entity type in the association. Minimum: 0 Maximum: 36
created_at	String	Time when the association was created.
updated_at	String	Time when the association was last updated.
external	Boolean	Whether the principle is in the same organization as the resource owner.
resource_share_id	String	ID of the resource share. Minimum: 0 Maximum: 36
resource_share_name	String	Name of the resource share. Minimum: 0 Maximum: 64
status	String	Status of the association. Minimum: 0 Maximum: 36
status_message	String	Description of the status to the association. Minimum: 0 Maximum: 65535

Example Requests

Disassociating specified principals or resources from a resource share

```
POST https://{hostname}/v1/resource-shares/{resource_share_id}/disassociate
```

```
{
  "principals" : [ "string" ],
  "resource_urns" : [ "string" ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_share_associations": [ {
    "associated_entity": "string",
    "association_type": "string",
    "created_at": "2022-08-22T11:53:04.912Z",
    "updated_at": "2022-08-22T11:53:04.912Z",
    "external": true,
    "resource_share_id": "string",
    "resource_share_name": "string",
    "status": "string",
    "status_message": "string"
  } ]
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.3.3 Searching for Associated Principals and Resources

Function

This API is used to search for the principals and resources associated with a resource share you created.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-share-associations/search

Request Parameters

Table 4-48 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 4-49 Request body parameters

Parameter	Mandatory	Type	Description
association_status	No	String	Status of the association. Minimum: 0 Maximum: 64
association_type	Yes	String	Association type (principal or resource).
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64
principal	No	String	Principal associated with the resource share. Minimum: 0 Maximum: 1500
resource_urn	No	String	URN of the resource associated with the resource share. Minimum: 0 Maximum: 1500
resource_share_ids	No	Array of strings	List of resource share IDs. Minimum: 0 Maximum: 36 Array Length: 1 - 100
resource_ids	No	Array of strings	List of resource IDs. Minimum: 1 Maximum: 1500 Array Length: 1 - 100

Response Parameters

Status code: 200

Table 4-50 Response body parameters

Parameter	Type	Description
resource_share_associations	Array of ResourceShareAssociation objects	List of association details. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-51 ResourceShareAssociation

Parameter	Type	Description
associated_entity	String	Associated entity. It can be the resource URN, account ID, URN of the root OU, or URN of another OU. Minimum: 0 Maximum: 1500
association_type	String	Entity type in the association. Minimum: 0 Maximum: 36
created_at	String	Time when the association was created.
updated_at	String	Time when the association was last updated.
external	Boolean	Whether the principle is in the same organization as the resource owner.
resource_share_id	String	ID of the resource share. Minimum: 0 Maximum: 36
resource_share_name	String	Name of the resource share. Minimum: 0 Maximum: 64
status	String	Status of the association. Minimum: 0 Maximum: 36
status_message	String	Description of the status to the association. Minimum: 0 Maximum: 65535

Table 4-52 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Searching for the principals and resources associated with a resource share

POST https://{hostname}/v1/resource-share-associations/search

```
{
  "association_status": "string",
  "association_type": "principal",
  "limit": 2000,
  "marker": "string",
  "principal": "string",
  "resource_urn": "string",
  "resource_share_ids": [ "string" ],
  "resource_ids": [ "string" ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_share_associations": [ {
    "associated_entity": "string",
    "association_type": "string",
    "created_at": "2022-09-19T02:10:16.960Z",
    "updated_at": "2022-09-19T02:10:16.960Z",
    "external": true,
    "resource_share_id": "string",
    "resource_share_name": "string",
    "status": "string",
    "status_message": "string"
  } ],
  "page_info": {
    "previous_marker": "string",
    "next_marker": "string",
    "current_count": 2000
  }
}
```


Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.4 Associated RAM Managed Permissions

4.4.1 Associating or Replacing RAM Managed Permissions

Function

This API is used to associate or replace the RAM managed permission for a resource type included in a resource share. You can have only one permission associated with each resource type in the resource share. You can associate a new RAM managed permission with a resource type only when there are currently no resources of that resource type in the resource share.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/{resource_share_id}/associate-permission

Table 4-53 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-54 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 4-55 Request body parameters

Parameter	Mandatory	Type	Description
replace	No	Boolean	Whether to use the specified permission to replace the existing permission associated with the resource share or to associate the specified permission for the current resource type with the resource share. Set this parameter to true to replace the current permission with the specified permission for the same resource type. Set this parameter to false to associate the specified permission with the current resource type. The default value is false. Only one permission can be associated with each resource type in the resource share. If the resource share already has a permission for the specified resource type and this parameter is set to false, an error is returned. This helps prevent accidental overwriting of the permission.
permission_id	Yes	String	ID of the RAM managed permission. Minimum: 0 Maximum: 36

Response Parameters

None

Example Requests

Associating or replacing the resource sharing permissions for a resource type in a resource share

```
POST https://{hostname}/v1/resource-shares/{resource_share_id}/associate-permission
{
  "permission_id" : "string",
  "replace" : true
}
```

Example Responses

None

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.4.2 Disassociating RAM Managed Permissions

Function

This API is used to disassociate RAM managed permissions from a resource share. The disassociation takes effect immediately after you call this API. You can disassociate RAM managed permissions for a resource type from a resource share only when there is no permission for that resource type in the resource share.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/{resource_share_id}/disassociate-permission

Table 4-56 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-57 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 4-58 Request body parameters

Parameter	Mandatory	Type	Description
permission_id	Yes	String	ID of the RAM managed permission. Minimum: 1 Maximum: 36

Response Parameters

None

Example Requests

Disassociating resource sharing permissions from a resource share

```
POST https://{hostname}/v1/resource-shares/{resource_share_id}/disassociate-permission
{
  "permission_id" : "string"
}
```

Example Responses

None

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.4.3 Getting Associated RAM Managed Permissions

Function

This API is used to get the details of RAM managed permissions associated with a resource share.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/resource-shares/{resource_share_id}/associated-permissions

Table 4-59 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Table 4-60 Query Parameters

Parameter	Mandatory	Type	Description
permission_name	No	String	Name of the RAM managed permission. Minimum: 1 Maximum: 64

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64

Request Parameters

Table 4-61 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 4-62 Response body parameters

Parameter	Type	Description
associated_permissions	Array of AssociatedPermission objects	List of RAM managed permissions associated with the resource share. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-63 AssociatedPermission

Parameter	Type	Description
permission_id	String	Permission ID. Minimum: 0 Maximum: 36
permission_name	String	Name of the RAM managed permission. Minimum: 0 Maximum: 128
resource_type	String	Resource type to which the permission applies. Minimum: 0 Maximum: 100
status	String	Status of the permission. Minimum: 0 Maximum: 36
created_at	String	Time when the permission was created.
updated_at	String	Time when the permission was last updated.

Table 4-64 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Getting the details of resource sharing permissions associated with a resource share

```
GET https://{hostname}/v1/resource-shares/{resource_share_id}/associated-permissions
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "associated_permissions": [ {
    "permission_id": "string",
    "permission_name": "string",
    "resource_type": "string",
    "status": "string",
    "created_at": "2022-09-19T02:12:30.736Z",
    "updated_at": "2022-09-19T02:12:30.736Z"
  } ],
  "page_info": {
    "previous_marker": "string",
    "next_marker": "string",
    "current_count": 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.5 Tag Management

4.5.1 Querying Tags Used by Resource Shares

Function

This API is used to query the list of tags used by resource shares.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/resource-shares/tags

Table 4-65 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64

Request Parameters

Table 4-66 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 4-67 Response body parameters

Parameter	Type	Description
tags	Array of TagDTO objects	List of one or more tag key-value pairs. The tag key cannot be left blank or be an empty string. The tag value must be specified and can be an empty string. Array Length: 0 - 2000
page_info	PageInfoMarkerByKey object	Pagination details.

Table 4-68 TagDTO

Parameter	Type	Description
key	String	Identifier or name of the tag key. You can set the tag value to an empty string, but cannot set it to null. Minimum: 1 Maximum: 128
values	Array of strings	List of values. Each value can contain a maximum of 255 characters. Minimum: 0 Maximum: 255 Array Length: 0 - 2000

Table 4-69 PageInfoMarkerByKey

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Querying the list of tags used by a resource share

```
GET https://{hostname}/v1/resource-shares/tags
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "tags": [ {
    "key": "string",
    "values": [ "string" ]
  } ],
  "page_info": {
    "previous_marker": "string",
```

```
"next_marker" : "string",  
"current_count" : 2000  
}  
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.5.2 Querying Resource Shares by Tag

Function

This API is used to query resource shares by tag.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/resource-instances/filter

Table 4-70 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 1000 Default: 1000
offset	No	String	Pagination marker.

Request Parameters

Table 4-71 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 4-72 Request body parameters

Parameter	Mandatory	Type	Description
without_any_tag	No	Boolean	If this parameter is set to true, all resources without tags are queried.
tags	No	Array of TagFilter objects	A maximum of 20 keys can be queried at a time, and each key can contain a maximum of 20 values. The structure body must be included. The tag key cannot be left blank or be an empty string. Each tag key must be unique, and each tag value for a key must be unique. Resources that contain all keys and one or multiple values listed in tags will be found and returned. If no tag filtering criteria is specified, all data is returned. Array Length: 0 - 20
matches	No	Array of Match objects	Matching field used for attaching tags to resource shares. Array Length: 0 - 20

Table 4-73 TagFilter

Parameter	Mandatory	Type	Description
key	Yes	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
values	No	Array of strings	List of values for the tag key. Minimum: 0 Maximum: 255 Array Length: 0 - 20

Table 4-74 Match

Parameter	Mandatory	Type	Description
key	Yes	String	Key. The value can be policy, organizational-unit, or account. Minimum: 0 Maximum: 128
value	Yes	String	Value. Each value can contain a maximum of 255 characters. Minimum: 0 Maximum: 255

Response Parameters

Status code: **200**

Table 4-75 Response body parameters

Parameter	Type	Description
resources	Array of ResourceDTO objects	List of resources. Array Length: 0 - 1000
total_count	Integer	Total number of records.

Table 4-76 ResourceDTO

Parameter	Type	Description
resource_id	String	Resource ID.
resource_name	String	Resource name.
tags	Array of Tag objects	List of resource tags. Array Length: 0 - 20
resource_detail	Object	Resource details.

Table 4-77 Tag

Parameter	Type	Description
key	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	String	Tag value. You can set the tag value to an empty string but not to null. Minimum: 0 Maximum: 255

Example Requests

Querying resource shares by tag

POST https://{hostname}/v1/resource-shares/resource-instances/filter

```
{
  "without_any_tag": true,
  "tags": [ {
    "key": "string",
    "values": [ "string" ]
  } ],
  "matches": [ {
    "key": "string",
    "value": "string"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resources": [ {
    "resource_id": "string",
```

```
"resource_name" : "string",  
"resource_detail" : { },  
"tags" : [ {  
  "key" : "string",  
  "value" : "string"  
} ]  
},  
"total_count" : 0  
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.5.3 Querying Number of Resource Shares by Tag

Function

This API is used to query the number of resource shares by tag.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/resource-instances/count

Request Parameters

Table 4-78 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 4-79 Request body parameters

Parameter	Mandatory	Type	Description
without_any_tag	No	Boolean	If this parameter is set to true, all resources without tags are queried.
tags	No	Array of TagFilter objects	A maximum of 20 keys can be queried at a time, and each key can contain a maximum of 20 values. The structure body must be included. The tag key cannot be left blank or be an empty string. Each tag key must be unique, and each tag value for a key must be unique. Resources that contain all keys and one or multiple values listed in tags will be found and returned. If no tag filtering criteria is specified, all data is returned. Array Length: 0 - 20
matches	No	Array of Match objects	Matching field used for attaching tags to resource shares. Array Length: 0 - 20

Table 4-80 TagFilter

Parameter	Mandatory	Type	Description
key	Yes	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
values	No	Array of strings	List of values for the tag key. Minimum: 0 Maximum: 255 Array Length: 0 - 20

Table 4-81 Match

Parameter	Mandatory	Type	Description
key	Yes	String	Key. The value can be policy, organizational-unit, or account. Minimum: 0 Maximum: 128
value	Yes	String	Value. Each value can contain a maximum of 255 characters. Minimum: 0 Maximum: 255

Response Parameters

Status code: 200

Table 4-82 Response body parameters

Parameter	Type	Description
total_count	Integer	Total number of records. Minimum: 0 Maximum: 1000000

Example Requests

Querying the number of resource shares by tag

```
POST https://{hostname}/v1/resource-shares/resource-instances/count
```

```
{
  "without_any_tag": true,
  "tags": [ {
    "key": "string",
    "values": [ "string" ]
  } ],
  "matches": [ {
    "key": "string",
    "value": "string"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "total_count": 100
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.5.4 Adding Tags to Resource Shares

Function

This API is used to add tags to resource shares.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/{resource_share_id}/tags/create

Table 4-83 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-84 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 4-85 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of Tag objects	List of one or more key-value pairs. The tag key must be specified and cannot be an empty string. The tag value must be specified and can be an empty string. Array Length: 1 - 20

Table 4-86 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	Yes	String	Tag value. You can set the tag value to an empty string but not to null. Minimum: 0 Maximum: 255

Response Parameters

None

Example Requests

Adding tags to a resource share

POST https://{hostname}/v1/resource-shares/{resource_share_id}/tags/create

```
{
  "tags": [ {
    "key": "string",
    "value": "string"
  } ]
}
```

Example Responses

None

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.5.5 Deleting Tags from Resource Shares

Function

This API is used to delete the specified tags from resource shares.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-shares/{resource_share_id}/tags/delete

Table 4-87 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_id	Yes	String	ID of the resource share. Minimum: 0 Maximum: 36

Request Parameters

Table 4-88 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 4-89 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of Untag objects	List of one or more key-value pairs. The tag key must be specified and cannot be an empty string. The tag value must be specified and can be an empty string. Array Length: 1 - 20

Table 4-90 Untag

Parameter	Mandatory	Type	Description
key	Yes	String	Identifier or name of the tag key. Minimum: 1 Maximum: 128
value	No	String	Tag value. You can set the tag value to an empty string but not null. Minimum: 0 Maximum: 255

Response Parameters

None

Example Requests

Deleting tags from a resource share

POST https://{hostname}/v1/resource-shares/{resource_share_id}/tags/delete

```
{
  "tags": [ {
    "key": "string",
    "value": "string"
  } ]
}
```

Example Responses

None

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.6 Shared Resources

4.6.1 Searching for Shared Resources

Function

This API is used to search for the resources that you share or are shared with you.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/shared-resources/search

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.

Table 4-92 Request body parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64
principal	No	String	Principal associated with the resource share. Minimum: 1 Maximum: 1500
resource_ids	No	Array of strings	List of resource IDs. Minimum: 1 Maximum: 1500 Array Length: 1 - 100
resource_urns	No	Array of strings	List of resource URNs. Minimum: 0 Maximum: 1500 Array Length: 1 - 100
resource_owner	Yes	String	Resource owner (self or other-accounts) of the resource share.
resource_share_ids	No	Array of strings	List of resource share IDs. Minimum: 0 Maximum: 36 Array Length: 1 - 100
resource_region	No	String	Region where the resource is located. Minimum: 0 Maximum: 255
resource_type	No	String	Resource type. Minimum: 0 Maximum: 64

Response Parameters

Status code: 200

Table 4-93 Response body parameters

Parameter	Type	Description
shared_resources	Array of SharedResource objects	List of shared resources. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-94 SharedResource

Parameter	Type	Description
resource_urn	String	Uniform resource name of the resource. Minimum: 0 Maximum: 1500
created_at	String	Time when the resource was associated with the resource share.
updated_at	String	Time when the resource share was last updated.
resource_type	String	Resource type. Minimum: 0 Maximum: 100
resource_share_id	String	ID of the resource share associated with the resource. Minimum: 0 Maximum: 36
status	String	Status of the association. Minimum: 0 Maximum: 36

Table 4-95 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Searching for the resources that you share or are shared with you

POST https://{hostname}/v1/shared-resources/search

```
{
  "limit" : 2000,
  "marker" : "string",
  "principal" : "string",
  "resource_ids" : [ "string" ],
  "resource_urns" : [ "string" ],
  "resource_owner" : "self",
  "resource_share_ids" : [ "string" ],
  "resource_region" : "string",
  "resource_type" : "string"
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "shared_resources" : [ {
    "resource_urn" : "string",
    "created_at" : "2022-09-19T02:32:23.735Z",
    "updated_at" : "2022-09-19T02:32:23.735Z",
    "resource_type" : "string",
    "resource_share_id" : "string",
    "status" : "string"
  } ],
  "page_info" : {
    "previous_marker" : "string",
    "next_marker" : "string",
    "current_count" : 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.7 Principals

4.7.1 Searching for Principals

Function

This API is used to search for the principals that have access to shared resources.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/shared-principals/search

Request Parameters

Table 4-96 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 4-97 Request body parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64
principals	No	Array of strings	Principal associated with the resource share. Maximum: 1500 Array Length: 0 - 100
resource_urn	No	String	URN of the resource. Minimum: 0 Maximum: 1500
resource_owner	Yes	String	Resource owner (self or other-accounts) of the resource share.
resource_share_ids	No	Array of strings	List of resource share IDs. Minimum: 0 Maximum: 36 Array Length: 1 - 100

Response Parameters

Status code: 200

Table 4-98 Response body parameters

Parameter	Type	Description
shared_principals	Array of SharedPrincipal objects	List of principal details. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-99 SharedPrincipal

Parameter	Type	Description
resource_share_id	String	ID of the resource share associated with the principal. Minimum: 0 Maximum: 36
id	String	Account ID of the principal or URN of the resource. Minimum: 0 Maximum: 1500
created_at	String	Time when the principal was associated with the resource share.
updated_at	String	Time when the resource share was last updated.

Table 4-100 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Searching for the principals that have access to shared resources

POST https://{hostname}/v1/shared-principals/search

```
{
  "limit" : 2000,
  "marker" : "string",
  "principals" : [ "string" ],
  "resource_urn" : "string",
  "resource_owner" : "self",
  "resource_share_ids" : [ "string" ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "shared_principals": [ {
    "resource_share_id": "string",
    "id": "string",
    "created_at": "2022-09-19T02:33:13.496Z",
    "updated_at": "2022-09-19T02:33:13.496Z"
  } ],
  "page_info": {
    "previous_marker": "string",
    "next_marker": "string",
    "current_count": 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.8 Resource Sharing Invitations

4.8.1 Accepting a Resource Sharing Invitation

Function

This API is used to accept a resource sharing invitation from other accounts.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-share-invitations/{resource_share_invitation_id}/accept

Table 4-101 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_invitation_id	Yes	String	ID of the resource sharing invitation. Minimum: 0 Maximum: 36

Request Parameters

Table 4-102 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 4-103 Response body parameters

Parameter	Type	Description
resource_share_invitation	ResourceShareInvitation object	Details about the resource sharing invitation.

Table 4-104 ResourceShareInvitation

Parameter	Type	Description
receiver_account_id	String	ID of the account that receives the resource sharing invitation. Minimum: 0 Maximum: 36
resource_share_id	String	ID of the resource share. Minimum: 0 Maximum: 36

Parameter	Type	Description
resource_share_name	String	Name of the resource share. Minimum: 0 Maximum: 64
resource_share_invitation_id	String	ID of the resource sharing invitation. Minimum: 0 Maximum: 36
sender_account_id	String	ID of the account that sends a resource sharing invitation. Minimum: 0 Maximum: 36
status	String	Status of the resource sharing invitation. Minimum: 0 Maximum: 64
created_at	String	Time when the invitation was created.
updated_at	String	Time when the invitation was last updated.

Example Requests

Accepting resource sharing invitations from other accounts

POST https://{hostname}/v1/resource-share-invitations/{resource_share_invitation_id}/accept

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_share_invitation": {
    "receiver_account_id": "string",
    "resource_share_id": "string",
    "resource_share_name": "string",
    "resource_share_invitation_id": "string",
    "sender_account_id": "string",
    "status": "string",
    "created_at": "2022-08-22T12:28:26.631Z",
    "updated_at": "2022-08-22T12:28:26.631Z"
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.8.2 Rejecting a Resource Sharing Invitation

Function

This API is used to reject a resource sharing invitation from other accounts.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-share-invitations/{resource_share_invitation_id}/reject

Table 4-105 Path Parameters

Parameter	Mandatory	Type	Description
resource_share_invitation_id	Yes	String	ID of the resource sharing invitation. Minimum: 0 Maximum: 36

Request Parameters

Table 4-106 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 4-107 Response body parameters

Parameter	Type	Description
resource_share_invitation	ResourceShareInvitation object	Details about the resource sharing invitation.

Table 4-108 ResourceShareInvitation

Parameter	Type	Description
receiver_account_id	String	ID of the account that receives the resource sharing invitation. Minimum: 0 Maximum: 36
resource_share_id	String	ID of the resource share. Minimum: 0 Maximum: 36
resource_share_name	String	Name of the resource share. Minimum: 0 Maximum: 64
resource_share_invitation_id	String	ID of the resource sharing invitation. Minimum: 0 Maximum: 36
sender_account_id	String	ID of the account that sends a resource sharing invitation. Minimum: 0 Maximum: 36
status	String	Status of the resource sharing invitation. Minimum: 0 Maximum: 64
created_at	String	Time when the invitation was created.
updated_at	String	Time when the invitation was last updated.

Example Requests

Rejecting resource sharing invitations from other accounts

```
POST https://{hostname}/v1/resource-share-invitations/{resource_share_invitation_id}/reject
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_share_invitation" : {
    "receiver_account_id" : "string",
    "resource_share_id" : "string",
    "resource_share_name" : "string",
    "resource_share_invitation_id" : "string",
    "sender_account_id" : "string",
    "status" : "string",
    "created_at" : "2022-08-22T12:29:00.460Z",
    "updated_at" : "2022-08-22T12:29:00.460Z"
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.8.3 Searching for a Resource Sharing Invitation

Function

This API is used to search for a resource sharing invitation based on certain search criteria.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/resource-share-invitations/search

Request Parameters

Table 4-109 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 4-110 Request body parameters

Parameter	Mandatory	Type	Description
resource_share_ids	No	Array of strings	List of resource share IDs. Minimum: 0 Maximum: 36 Array Length: 0 - 100
resource_share_invitation_ids	No	Array of strings	List of resource sharing invitation IDs. Minimum: 0 Maximum: 36 Array Length: 0 - 100
status	No	String	Status of the resource sharing invitation. Minimum: 0 Maximum: 64
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50

Parameter	Mandatory	Type	Description
marker	No	String	Paging location marker. The query starts from the next piece of data indexed by this parameter. You do not need to specify this parameter when you query resources on the first page. When you query resources on subsequent pages, set this parameter to the location returned in the response body for the previous query. Minimum: 1 Maximum: 64

Response Parameters

Status code: **200**

Table 4-111 Response body parameters

Parameter	Type	Description
resource_share_invitations	Array of ResourceShareInvitation objects	Details about the resource sharing invitation. Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-112 ResourceShareInvitation

Parameter	Type	Description
receiver_account_id	String	ID of the account that receives the resource sharing invitation. Minimum: 0 Maximum: 36
resource_share_id	String	ID of the resource share. Minimum: 0 Maximum: 36

Parameter	Type	Description
resource_share_name	String	Name of the resource share. Minimum: 0 Maximum: 64
resource_share_invitation_id	String	ID of the resource sharing invitation. Minimum: 0 Maximum: 36
sender_account_id	String	ID of the account that sends a resource sharing invitation. Minimum: 0 Maximum: 36
status	String	Status of the resource sharing invitation. Minimum: 0 Maximum: 64
created_at	String	Time when the invitation was created.
updated_at	String	Time when the invitation was last updated.

Table 4-113 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Searching for resource sharing invitations based on certain search criteria

POST https://{hostname}/v1/resource-share-invitations/search

```
{
  "resource_share_ids" : [ "string" ],
  "resource_share_invitation_ids" : [ "string" ],
  "status" : "string",
```

```
"limit" : 2000,  
"marker" : "string"  
}
```

Example Responses

Status code: 200

Request succeeded.

```
{  
  "resource_share_invitations": [ {  
    "receiver_account_id": "string",  
    "resource_share_id": "string",  
    "resource_share_name": "string",  
    "resource_share_invitation_id": "string",  
    "sender_account_id": "string",  
    "status": "string",  
    "created_at": "2022-09-23T06:43:43.999Z",  
    "updated_at": "2022-09-23T06:43:43.999Z"  
  } ],  
  "page_info": {  
    "previous_marker": "string",  
    "next_marker": "string",  
    "current_count": 2000  
  }  
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.9 Sharing with Organizations

4.9.1 Checking Whether Sharing with Organizations Is Enabled

Function

This API is used to check whether sharing with Organizations is enabled.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/organization-share

Request Parameters

Table 4-114 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 4-115 Response body parameters

Parameter	Type	Description
enabled	Boolean	Whether sharing with Organizations is enabled. If the value is true, sharing with Organizations is enabled. The default value is false.

Example Requests

Checking whether sharing with Organizations is enabled

```
GET https://{hostname}/v1/organization-share
```

Example Responses

Status code: 200

Request succeeded.

```
{  
  "enabled" : true  
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.9.2 Enabling Sharing with Organizations

Function

This API is used to enable sharing with Organizations.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/organization-share/enable

Request Parameters

Table 4-116 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

Enabling sharing with Organizations

```
POST https://{hostname}/v1/organization-share/enable
```


Example Responses

None

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.9.3 Disabling Sharing with Organizations

Function

This API is used to disable sharing with Organizations.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

POST /v1/organization-share/disable

Request Parameters

Table 4-117 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

Disabling sharing with Organizations

```
POST https://{hostname}/v1/organization-share/disable
```

Example Responses

None

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.10 Other operations

4.10.1 Querying Resource Sharing Quotas

Function

This API is used to query resource sharing quotas in the current account.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

```
GET /v1/resource-shares/quotas
```

Request Parameters

Table 4-118 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 4-119 Response body parameters

Parameter	Type	Description
quotas	QuotaResourcesDto object	List of resource sharing quotas.

Table 4-120 QuotaResourcesDto

Parameter	Type	Description
resources	Array of Quotas objects	Quota information.

Table 4-121 Quotas

Parameter	Type	Description
type	String	Quota type. resource_share indicates the number of resource shares allowed to be created by the current account; resource_association indicates the number of resources allowed to be associated with a resource share; principal_association indicates the number of principals allowed to be associated with a resource share; permission_association indicates the number of permissions allowed to be associated with a resource share; tag_association indicates the number of tags allowed to be associated with a resource share. Minimum: 1 Maximum: 64
quota	Integer	Total number of quotas.
min	Integer	Minimum quota.
max	Integer	Maximum quota.
used	Integer	Used quotas.

Example Requests

Querying resource sharing quotas for the current account

```
GET https://{hostname}/v1/resource-shares/quotas
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "quotas" : {
    "resources" : [ {
      "type" : "string",
      "quota" : 0,
      "min" : 0,
      "max" : 1,
      "used" : 0
    } ]
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

4.10.2 Querying Resource Type

Function

This API is used to query the resource types and applicable regions of interconnected cloud services.

Debugging

You can debug this API through automatic authentication in [API Explorer](#) or use the SDK sample code generated by API Explorer.

URI

GET /v1/resource-types

Table 4-122 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of entries that can be displayed on a page. Minimum: 1 Maximum: 2000 Default: 50
marker	No	String	Page marker. Minimum: 1 Maximum: 64

Request Parameters

Table 4-123 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 4-124 Response body parameters

Parameter	Type	Description
resource_types	Array of ResourceTypesSummary objects	List of details about cloud service resources Array Length: 0 - 2000
page_info	PageInfo object	Pagination details.

Table 4-125 ResourceTypesSummary

Parameter	Type	Description
region_id	String	Name of the region to which the resources belong Minimum: 1 Maximum: 255
resource_type	String	Name of the resource type Minimum: 1 Maximum: 64

Table 4-126 PageInfo

Parameter	Type	Description
previous_marker	String	Marker of the previous page. Minimum: 1 Maximum: 64
next_marker	String	Marker of the next page. Minimum: 1 Maximum: 64
current_count	Integer	Page size. Minimum: 1 Maximum: 2000

Example Requests

Getting the resource types and applicable regions of interconnected cloud services

GET https://{hostname}/v1/resource-types

Example Responses

Status code: 200

Request succeeded.

```
{
  "resource_types": [ {
    "region_id": "string",
    "resource_type": "string"
  } ],
  "page_info": {
    "previous_marker": "string",
    "next_marker": "string",
    "current_count": 2000
  }
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See [Error Codes](#).

5 Permissions and Supported Actions

5.1 Introduction

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

New IAM users do not have any permissions assigned by default. You need to first add them to one or more groups and attach policies or roles to these groups. The users then inherit permissions from the groups and can perform specified operations on cloud services based on the permissions they have been assigned.

You can grant users permissions by using **roles** and **policies**. Roles are provided by IAM to define service-based permissions that match users' job responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

If you want to allow or deny the access to an API, use policy-based authorization.

If you use IAM users in your account to call an API, the IAM users must be granted the required permissions. The required permissions are determined by the actions supported by the API. Only users with the policies allowing for those actions can call the API successfully. For example, if an IAM user wants to query RAM using an API, the user must have been granted permissions that allow the **ram:permissions:list** action.

Supported Actions

RAM provides system-defined policies that can be directly used in IAM. You can also create custom policies to supplement system-defined policies for more refined access control. 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 in a custom policy
- **Dependencies:** actions which a specific action depends on. When allowing an action for a user, you also need to allow any existing action dependencies for that user.
- **IAM projects/Enterprise projects:** the authorization scope of a custom policy. A custom policy can be applied to IAM projects or enterprise projects or both. Policies that contain actions for both IAM and enterprise projects can be used and applied for both IAM and Enterprise Management. Policies that contain actions only for IAM projects can be used and applied to IAM only. For details about the differences between IAM and enterprise projects, see [Differences Between IAM and Enterprise Projects](#).

 **NOTE**

The check mark (√) indicates that an action takes effect. The cross mark (x) indicates that an action does not take effect.

5.2 Actions

Table 5-1 RAM actions

Permission	API	Action	IAM Project	Enterprise Project
Listing RAM managed permissions	GET /v1/permissions	ram:permissions:list	×	×
Getting the details about RAM managed permissions	GET /v1/permissions/{permission_id}	ram:permissions:get	×	×
Getting all versions of permissions	GET /v1/permissions/{permission_id}/versions	ram:permissions:listVersions	×	×
Creating a resource share	POST /v1/resource-shares	ram:resourceShares:create	×	×
Searching for resource shares	POST /v1/resource-shares/search	ram:resourceShares:search	×	×

Permission	API	Action	IAM Project	Enterprise Project
Updating properties of a resource share	PATCH /v1/resource-shares/{resource_share_id}	ram:resourceShares:update	×	×
Deleting a resource share	DELETE /v1/resource-shares/{resource_share_id}	ram:resourceShares:delete	×	×
Associating principals and resources with a resource share	POST /v1/resource-shares/{resource_share_id}/associate	ram:resourceShares:associate	×	×
Disassociating principals and resources from a resource share	POST /v1/resource-shares/{resource_share_id}/disassociate	ram:resourceShares:disassociate	×	×
Searching for associated principals and resources	POST /v1/resource-share-associations/search	ram:resourceShares:searchResourceShareAssociations	×	×
Associating or replacing RAM managed permissions	POST /v1/resource-shares/{resource_share_id}/associate-permission	ram:resourceShares:associatePermission	×	×
Disassociating RAM managed permissions	POST /v1/resource-shares/{resource_share_id}/disassociate-permission	ram:resourceShares:disassociatePermission	×	×

Permission	API	Action	IAM Project	Enterprise Project
Listing associated RAM managed permissions	GET /v1/resource-shares/{resource_share_id}/associated-permissions	ram:resourceShares:listAssociatedPermissions	×	×
Listing used tags	GET /v1/resource-shares/tags	ram:resourceShares:listTags	×	×
Listing resource shares by tag	POST /v1/resource-shares/resource-instances/filter	ram:resourceShares:listResourceSharesByTag	×	×
Listing the number of resource shares by tag	POST /v1/resource-shares/resource-instances/count	ram:resourceShares:searchResourceShareCountByTag	×	×
Adding a tag to a resource share	POST /v1/resource-shares/{resource_share_id}/tags/create	ram:resourceShares:tag	×	×
Deleting a tag from a resource share	POST /v1/resource-shares/{resource_share_id}/tags/delete	ram:resourceShares:untag	×	×
Searching for shared resources	POST /v1/shared-resources/search	ram:sharedResources:search	×	×
Searching for principals	POST /v1/shared-principals/search	ram:sharedPrincipals:search	×	×
Accepting a resource sharing invitation	POST /v1/resource-share-invitations/{resource_share_invitation_id}/accept	ram:resourceShareInvitations:accept	×	×
Rejecting a resource sharing invitation	POST /v1/resource-share-invitations/{resource_share_invitation_id}/reject	ram:resourceShareInvitations:reject	×	×

Permission	API	Action	IAM Project	Enterprise Project
Searching for a resource sharing invitation	POST /v1/resource-share-invitations/search	ram:resourceShareInvitations:search	×	×
Enabling sharing with Organizations	POST /v1/resource-shares/enable-sharing-with-organization	ram:resourceShares:enableSharingWithOrganization	×	×
Disabling sharing with Organizations	POST /v1/resource-shares/disable-sharing-with-organization	ram:resourceShares:disableSharingWithOrganization	×	×
Checking whether sharing with Organizations is enabled	POST /v1/resource-shares/search-enabled-sharing-with-organization	ram:resourceShares:searchEnableSharingWithOrganization	×	×
Listing RAM quotas	GET /v1/resource-shares/quotas	ram:resourceShares:listQuota	×	×
Querying resource types of cloud services	GET /v1/resource-types	ram:resourceTypes:list	×	×

6 Appendixes

6.1 Status Codes

Table 6-1 Status codes

Status Code	Message Title	Description
100	Continue	The client should continue with its request. This interim response is used to inform the client that the initial part of the request has been received and has not yet been rejected by the server.
101	Switching Protocols	The requester has asked the server to switch protocols and the server has agreed to do so. The protocol should be switched only when it is advantageous to do so. For example, switching to a newer version of HTTP is advantageous over older versions.
201	Created	The request has been fulfilled and resulted in a new resource being created.
202	Accepted	The request has been accepted for processing, but the processing has not been completed.
203	Non-Authoritative Information	The server successfully processed the request, but is returning information that may be from another source.
204	NoContent	The server successfully processed the request and is not returning any content. The status code is returned in response to an HTTP OPTIONS request.

Status Code	Message Title	Description
205	Reset Content	The server successfully processed the request, but is not returning any content.
206	Partial Content	The server has fulfilled the partial GET request for the resource.
300	Multiple Choices	There are multiple options for the resource from which the client may choose. For example, this code could be used to present a list of resource characteristics and addresses from which the client such as a browser may choose.
301	Moved Permanently	The requested resource has been assigned a new permanent URI and any future references to this resource should use one of the returned URIs.
302	Found	The requested resource resides temporarily under a different URI.
303	See Other	The response to the request can be found under a different URI and should be retrieved using a GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, it does not return any resources.
305	Use Proxy	The requested resource must be accessed through a proxy.
306	Unused	This HTTP status code is no longer used.
400	BadRequest	The request could not be understood by the server due to malformed syntax. The client should not repeat the request without modifications.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid. Check the username and password.
402	Payment Required	This status code is reserved for future use.
403	Forbidden	The server understood the request, but is refusing to fulfill it. The client should not repeat the request without modifications.
404	NotFound	The requested resource cannot be found. The client should not repeat the request without modifications.

Status Code	Message Title	Description
405	MethodNotAllowed	The method specified in the request is not allowed for the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server cannot fulfill the request based on the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The client does not produce a request within the time that the server was prepared to wait. The client may repeat the request without modifications at any later time.
409	Conflict	The request could not be completed due to a conflict with the current state of the resource. This status code indicates that the resource that the client attempts to create already exists, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource is no longer available. The requested resource has been deleted permanently.
411	Length Required	The server refuses to process the request without a defined Content-Length.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The server is refusing to process a request because the request entity is larger than the server is willing or able to process. The server may close the connection to prevent the client from continuing the request. If the condition is temporary, the server should include a Retry-After header field to indicate that it is temporary and after what time the client may try again.
414	Request-URI Too Large	The server is refusing to service the request because the request URI is longer than the server is willing to interpret.

Status Code	Message Title	Description
415	Unsupported Media Type	The server is refusing to service the request because the entity of the request is in a format not supported by the requested resource for the requested method.
416	Requested range not satisfiable	The requested range is invalid.
417	Expectation Failed	The server fails to meet the requirements of the Expect request header field.
422	UnprocessableEntity	The request was well-formed but was unable to be followed due to semantic errors.
429	TooManyRequests	The client has sent more requests than its rate limit is allowed within a given amount of time, or the server has received more requests than it is able to process within a given amount of time. In this case, the client should repeat requests after the time specified in the Retry-After header of the response expires.
500	InternalServerError	The server encountered an unexpected condition which prevented it from fulfilling the request.
501	Not Implemented	The server does not support the functionality required to fulfill the request.
502	Bad Gateway	The server, while acting as a gateway or proxy, received an invalid response from the upstream server it accessed in attempting to fulfill the request.
503	ServiceUnavailable	The requested service is unavailable. The client should not repeat the request without modifications.
504	ServerTimeout	The request cannot be fulfilled within a given amount of time. The response will reach the client only if the request carries a timeout parameter.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

6.2 Error Codes

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in [API Gateway Error Codes](#).

Status Code	Error Code	Error Message	Description	Solution
400	RAM.1001	The domain id in the header or the token is invalid.	Invalid account ID or token.	Check whether the account ID is correct.
400	RAM.1002	The domain id %s in the header is invalid.	Invalid account ID.	Check whether the account ID is correct.
400	RAM.1003	The token is invalid.	Invalid token.	Contact technical support.
400	RAM.1004	The account id is invalid.	Invalid account ID.	Check whether the account ID of the principal is correct.
400	RAM.1005	The share principal %s cannot be the owner.	Resource owner not allowed to act as principal.	Check whether the principal is correctly configured.
400	RAM.1006	The principal is invalid for duplicate.	Duplicate principals.	Delete the duplicate principals.
400	RAM.1007	The resource urn is invalid for duplicate.	Duplicate URNs of shared resources.	Delete the duplicate URNs.
400	RAM.1008	Bad request for checking permission.	Insufficient permissions for the shared resource.	Check whether the RAM permission has been granted to the IAM user.
400	RAM.1009	The resource type %s has no permission.	No permissions for the shared resource.	Grant the permissions to access the shared resource.
400	RAM.1010	The domain id %s does not own the resource urn %s.	Account ID and URN mismatched.	Check whether the account ID matches the URN.

Status Code	Error Code	Error Message	Description	Solution
400	RAM.1011	The %s num exceeds %s quota %s if add count %s.	Maximum quota exceeded.	Check the request parameters.
400	RAM.1012	The %s num exceeds the total quota %s if add count %s.	Maximum quota exceeded.	Check the request parameters.
400	RAM.1013	The organization management id %s has not enabled sharing.	Sharing with Organizations not enabled.	Enable sharing with Organizations.
400	RAM.1014	The cross organizations share cannot shared to the organizations with the principal %s.	Invalid cross-organization sharing.	Check whether the account of the principal belongs to another organization.
400	RAM.1015	The cross organizations share cannot shared to the organizations unit with the principal %s.	Invalid cross-organization sharing.	Check whether the account of the principal belongs to another organization.
400	RAM.1016	The cross organizations share cannot shared to the organizations root with the principal %s.	Invalid cross-organization sharing.	Check whether the account of the principal belongs to another organization.
404	RAM.1017	The resource share id %s is not found.	Invalid ID of a resource share.	Check whether the ID of the resource share is correct.

Status Code	Error Code	Error Message	Description	Solution
404	RAM.1018	The permission id %s is not found.	Invalid ID of the permission to access the shared resources.	Check whether the permission ID is correct.
400	RAM.1019	Please accept the open beta of resource access service.	The tenant has not applied for OBT.	Apply for OBT first.
404	RAM.1020	Get project failed.	Failed to get project information.	Check whether the account ID is correct.
404	RAM.1021	Get domain information failed.	Failed to obtain domain information.	Contact technical support.
404	RAM.1022	The principal %s is not exist.	Principal not found.	Check whether the principal is correctly configured.
404	RAM.1023	The principal %s is invalid.	Invalid principal.	Check whether the principal is correctly configured.
404	RAM.1024	The urn %s is invalid.	Invalid URN of the shared resources.	Check whether the URN is correct.
400	RAM.1025	Server SSL cert check failed.	Failed to check the SSL certificate.	Contact technical support.
400	RAM.1026	Query restricted list failed.	Failed to query the restricted list.	Contact technical support.
400	RAM.1027	Query frozen list failed.	Failed to query the frozen list.	Contact technical support.
400	RAM.1028	Bad request for notification.	Request error.	Contact technical support.

Status Code	Error Code	Error Message	Description	Solution
403	RAM.1029	The current operation has been restricted.	The current operation has been restricted.	Check whether the account is restricted.
403	RAM.1030	The current operation has been frozen.	The current operation has been frozen.	Check whether the account is frozen.
400	RAM.1031	Bad request for authorization header pattern.	Failed to verify the request header.	Contact technical support.
400	RAM.1032	Bad quota type.	Incorrect quota type.	Check whether the quota type is correct.
400	RAM.1033	Async job failed.	The job fails to be executed.	Contact technical support.
400	RAM.1034	Don't have lock	Failed to obtain a lock.	Contact technical support.
400	RAM.1035	Bad request for invalid context attributes.	Request error.	Contact technical support.
400	RAM.1036	Bad request for request proof.	Request error.	Contact technical support.
400	RAM.1037	Bad request for impersonate.	Request error.	Contact technical support.
400	RAM.1101	The status %s of the resource share with id %s is not active.	Resource share not activated.	Check the status of the resource share.
400	RAM.1102	There are resources in use in the resource share %s.	Resource already associated.	Check whether the resource has been already associated with another resource share.

Status Code	Error Code	Error Message	Description	Solution
400	RAM.1103	Some resources do not have or have multiple corresponding permissions.	No permissions or multiple permissions for the resource share.	Check whether the permissions are configured correctly for the resource share.
404	RAM.1104	Http header not found.	The HTTP request header is invalid.	Check whether the HTTP request header is valid.
400	RAM.1105	Bad request for invalid user profile.	The UserProfile parameter in the HTTP request header is invalid.	Check whether the UserProfile parameter in the HTTP request header is valid.
400	RAM.1106	Get domain tags failed.	Failed to obtain domain tags.	Contact technical support.
500	RAM.1107	Bad request for invalid token.	Failed to obtain the token of the OP account.	Contact technical support.
400	RAM.1108	Cannot share resource across sites between international account and national account.	Cross-site resource sharing is not supported.	Check whether the resource owner and principals register with the same site.
400	RAM.1109	Cannot share resource between op account and not-op account.	Resource sharing between OP accounts and other accounts is not supported.	Check whether the resource owner and principals are using accounts of the same type.

Status Code	Error Code	Error Message	Description	Solution
400	RAM.1110	Resource sharing not allowed between internal and external accounts.	Resource sharing not allowed between internal and external accounts.	Check whether the resource owner and principals are using accounts of the same type.
400	RAM.1201	The request body is empty.	Request parameters left blank.	Check the request parameters.
409	RAM.1202	Some principals have been associated to the resource share with id %s.	Same principal already associated.	Check whether the same principal is associated with multiple resource shares.
409	RAM.1203	Some resources have been associated to the resource share with id %s.	Resource already associated.	Check whether the same resource is associated with multiple resource shares.
400	RAM.1204	The status %s of the resource share is not active.	Resource share not activated.	Contact technical support.
400	RAM.1205	Get organization id error.	Failed to obtain the organization ID.	Check whether the account ID is correct.
400	RAM.1206	Get ancestors error.	Failed to obtain the upper-level organization.	Check whether the account ID is correct.

Status Code	Error Code	Error Message	Description	Solution
400	RAM.1207	Unable to disassociate the principal or the resource which is not exist or owned by you.	Disassociation not allowed for the resource or principal not in your organization or that does not exist.	Check whether the principal or resource is correctly configured.
400	RAM.1208	Unable to disassociate the principal because you belong to the organization.	Disassociation not allowed for the principal in your organization.	Check whether the principal belongs to your organization.
400	RAM.1209	Get organization path failed.	Failed to obtain the organization path of the tenant.	Contact technical support.
400	RAM.1301	The status %s of the resource share is not active.	Resource share not activated.	Contact technical support.
409	RAM.1302	The resource type %s has associated one permission.	Permission already associated with to the shared resource.	Check the request parameters.
409	RAM.1303	The resource type %s is in use in the resource share with id %s.	Failed to disassociate the permission from the in-use shared resource.	Stop using the shared resource and then disassociate the permission from the shared resource.
409	RAM.1701	The current status of the invitation is %s.	Abnormal status of the resource sharing invitation.	Check whether the status of the resource sharing invitation is correct.

Status Code	Error Code	Error Message	Description	Solution
404	RAM.1702	The resource share invitation with id %s is not found.	Invalid resource sharing invitation.	Check whether the resource sharing invitation has expired.
400	RAM.1801	The account id %s is not the management account of organization.	Account ID not being the management account.	Contact the organization administrator. Only the management account can perform this operation.
400	RAM.1802	Get organization info error.	Incorrect organization information.	Check whether organization information is correct.
400	RAM.1803	The sharing with organization is not enabled by management account %s.	Sharing with Organizations not enabled by the management account.	Contact the organization administrator to use the management account to enable sharing with Organizations.
501	RAM.1804	RAM is not support organization sharing function.	Sharing with Organizations not supported by RAM.	Contact technical support.
400	RAM.1901	Tag key is exist for resource share id.	The tag key is already used by the resource share.	Check whether there is a tag with the same name in the resource share.
400	RAM.1902	The tag key is duplicate.	The tag key is in duplicate.	Check whether the tag key is in duplicate.
400	RAM.1903	The tag value is duplicate.	The tag value is in duplicate.	Check whether the tag value is in duplicate.
400	RAM.1904	Failed to check the resource tag.	Failed to check the resource tag.	Contact technical support.

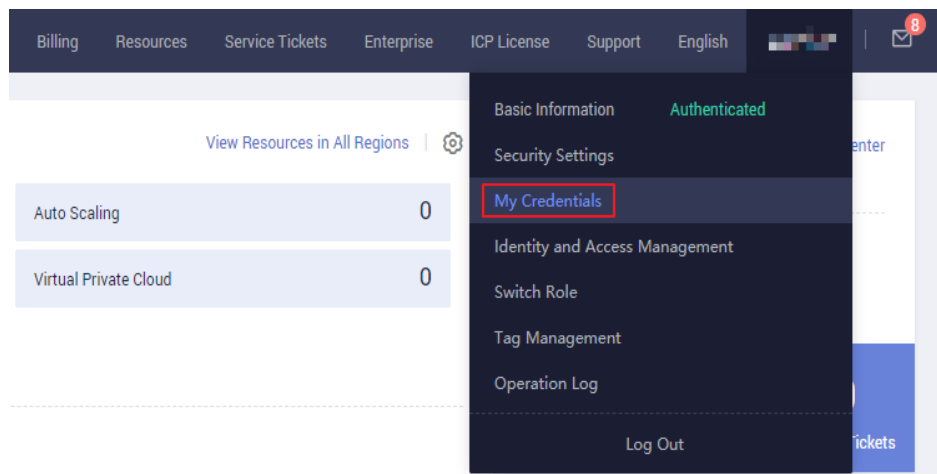
Status Code	Error Code	Error Message	Description	Solution
400	RAM.1905	The tag policy doesn't allow the specified value for the following tag key.	The tag policy does not allow the specified value for the following tag key.	Check whether the label complies with the tag policy.

6.3 Obtaining Information About Account, IAM User, Group, Project, Region, and Agency

Obtaining Account, IAM User, and Project Information

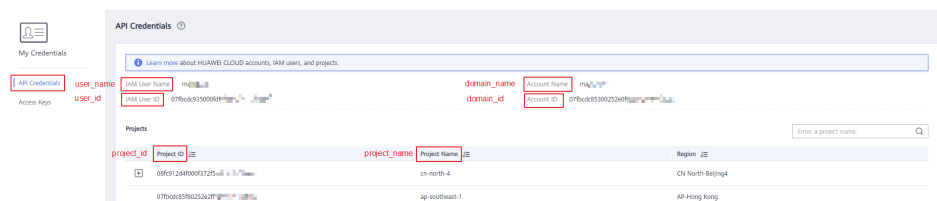
- Using the console
 - a. On the Huawei Cloud homepage, click **Console** in the upper right corner.
 - b. Hover over the username in the upper right corner and choose **My Credentials**.

Figure 6-1 My Credentials



- c. View the account name, account ID, username, user ID, project name, and project ID on the **API Credentials** page. The project ID varies depending on the region where the service is located.

Figure 6-2 Viewing the account, user, and project information



- **Calling an API**
 - For details about how to obtain a user ID, see [Listing IAM Users](#).
 - For details about how to obtain a project ID, see [Querying Project Information](#).

Obtaining User Group Information

Step 1 Log in to the IAM console, and choose **User Groups** in the navigation pane.

Step 2 Expand the details page of a user group and view the group name and ID.

----End

Obtaining Region Information

Step 1 Log in to the IAM console, and choose **Projects** in the navigation pane.

Step 2 The value in the **Project Name** column is the ID of the region to which the project belongs.

----End

Obtaining Agency Information

Step 1 Log in to the IAM console, and choose **Agencies** in the navigation pane.

Step 2 Hover over the desired agency. The name and ID of this agency are displayed in a dark pop-up box.

----End

7 Change History

Released On	Description
2024-03-15	This issue is the seventh official release, which incorporates the following changes: Resource Access Manager (RAM) is put into commercial use.
2023-12-30	This issue is the sixth official release, which incorporates the following changes: Added APIs for Getting all versions of permissions .
2023-08-30	This issue is the fifth official release, which incorporates the following change: Added APIs for Querying Resource Type .
2023-05-30	This issue is the fourth official release, which incorporates the following changes: Added APIs for Querying Resource Sharing Quotas .
2023-04-11	This issue is the third official release, which incorporates the following change: Added APIs for Tag Management .
2023-03-20	This issue is the second official issue, which incorporates the following change: Added some error codes.
2022-10-29	This issue is the first official release.